日　本　国　特　許　庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 Date of Application:	２００３年　８月２２日
出願番号 Application Number:	特願２００３－２９９１３５
[ST. 10/C]:	[ＪＰ２００３－２９９１３５]
出願人 Applicant(s):	株式会社リコー

２００３年１２月　５日

特許庁長官
Commissioner,
Japan Patent Office　今井康夫

出証番号　出証特２００３－３１００６８１

【書類名】　特許願
【整理番号】　0300145
【提出日】　平成15年 8月22日
【あて先】　特許庁長官殿
【国際特許分類】　G06F 12/00　520
【発明者】
　【住所又は居所】　東京都大田区中馬込１丁目３番６号　株式会社リコー内
　【氏名】　長谷川　雄史
【発明者】
　【住所又は居所】　東京都大田区中馬込１丁目３番６号　株式会社リコー内
　【氏名】　鈴木　明
【発明者】
　【住所又は居所】　東京都大田区中馬込１丁目３番６号　株式会社リコー内
　【氏名】　小出　雅巳
【特許出願人】
　【識別番号】　000006747
　【氏名又は名称】　株式会社リコー
【代理人】
　【識別番号】　100104190
　【弁理士】
　【氏名又は名称】　酒井　昭徳
【手数料の表示】
　【予納台帳番号】　041759
　【納付金額】　21,000円
【提出物件の目録】
　【物件名】　特許請求の範囲　1
　【物件名】　明細書　1
　【物件名】　図面　1
　【物件名】　要約書　1
　【包括委任状番号】　9810808

【書類名】特許請求の範囲
【請求項１】
動作プログラム提供者によって提供される、複数の動作プログラムおよび動作プログラム利用情報と、前記動作プログラム提供者または文書作成者によって作成されるコンテンツファイルおよびコンテンツファイル利用情報と、を含み構成されていることを特徴とする電子文書ファイル。
【請求項２】
前記複数の動作プログラムのうちの少なくとも一つは動作検証プログラムであり、該動作検証プログラムは、前記動作プログラム利用情報に基づき他の動作プログラムを動作させることを特徴とする請求項１に記載の電子文書ファイル。
【請求項３】
前記複数の動作プログラムのうち少なくとも一つはコンテンツ検証プログラムであり、該コンテンツ検証プログラムは、前記コンテンツファイル利用情報に基づきコンテンツファイルの利用を制限することを特徴とする請求項１または２に記載の電子文書ファイル。
【請求項４】
前記動作検証プログラムは前記動作プログラム利用情報に基づき他の動作プログラムを動作させ、該動作プログラムのうちの少なくとも一つはコンテンツ検証プログラムであり、該コンテンツ検証プログラムは前記コンテンツファイル利用情報に基づきコンテンツファイルの利用を制限することを特徴とする請求項２または３に記載の電子文書ファイル。
【請求項５】
前記動作検証プログラムは、前記動作プログラムおよび前記動作プログラム利用情報の改ざんを検証し、または前記動作プログラムの動作を制限することを特徴とする請求項２～４のいずれか一つに記載の電子文書ファイル。
【請求項６】
前記コンテンツ検証プログラムは、前記コンテンツファイルおよび前記コンテンツファイル利用情報の改ざんを検証し、または前記コンテンツファイルの利用を制限することを特徴とする請求項３～５のいずれか一つに記載の電子文書ファイル。
【請求項７】
前記動作検証プログラムによって前記動作プログラムの安全性が保証されない場合には、当該電子文書ファイルの使用者に前記動作プログラムの安全性が保証されていないことを認識させることを特徴とする請求項２～６のいずれか一つに記載の電子文書ファイル。
【請求項８】
前記動作プログラム、前記動作プログラム利用情報、前記コンテンツファイル、および前記コンテンツファイル利用情報は、単一ファイルにカプセル化されていることを特徴とする請求項１～７のいずれか一つに記載の電子文書ファイル。
【請求項９】
複数の動作プログラムおよび動作プログラム利用情報を読み込む読込手段と、コンテンツファイルおよびコンテンツファイル利用情報を作成するコンテンツファイル作成手段と、請求項１～８のいずれか一つに記載の電子文書ファイルに格納する各情報を作成する情報作成手段と、を含み構成されていることを特徴とする電子文書ファイル作成装置。
【請求項１０】
前記コンテンツファイル作成手段は、請求項１～８のいずれか一つに記載の電子文書ファイル内に格納されている複数の動作プログラムの一つを動作させることを特徴とする請求項９に記載の電子文書ファイル作成装置。
【請求項１１】
さらに前記コンテンツファイル作成手段により作成されたコンテンツファイルおよびコンテンツファイル利用情報に基づいて、動作プログラム提供者によって提供される複数の動作プログラムおよび動作プログラム利用情報を付加する動作プログラム付加手段を備えていることを特徴とする請求項９または１０に記載の電子文書ファイル作成装置。

【書類名】明細書
【発明の名称】電子文書ファイル、電子文書ファイル作成装置
【技術分野】
【０００１】
本発明は、電子文書の安全性を電子文書作成アプリケーションソフトが保証する電子文書認証システムに用いる電子文書ファイル、およびこの電子文書ファイルを作成する装置に関する。
【背景技術】
【０００２】
近年、コンピュータの発達によって、不特定多数の人々がデジタル情報で表現された電子文書を大量に作成するようになってきた。また、インターネット等に代表されるネットワークインフラの整備がされると共に、電子文書の作成者は有用な情報を多くの人々に伝えるため、作成した電子文書を多くの人々へ配布する機会が多くなってきている。また、配布される電子文書が増加しているため、電子文書を受信して閲覧する者が、当該電子文書の作成者を把握でない場合が増加している。これを悪用し、電子文書内に有害なデータを挿入し、閲覧者の意図と反してコンピュータの設定変更やコンピュータ内に保存されたデータの消去・改ざんなどを行う者が存在する。したがって、閲覧者は既知でない文書作成者から配布された電子文書を安心して閲覧することができないという不都合が生じる。
【０００３】
ところで、配布された電子文書がプレーンテキストデータだけで表現される場合には、有害なデータを挿入し難いために危険性は低い。しかし、現在普及している電子文書形態では、電子文書の表現力、編集能力などを豊かにする目的で電子文書内に特定の連続した操作を登録して自動化させるマクロ機能などが追加されているため、電子文書の危険性が高くなっている。
【０００４】
また、未知の文書作成者が電子文書内に有害なプログラムを挿入すること以外に、文書作成者が電子文書内に著作権法を侵害するコンテンツ内容を挿入することも問題として挙げられる。インターネットの発達によってネットワーク上に保存されている様々なコンテンツを容易に取得できることや、コンピュータハードウェア・ソフトウェアの発達によってコンテンツを容易に加工・編集できることにより、不特定多数の文書作成者が電子文書内には多様なコンテンツを挿入するようになった。このような背景により、作成した電子文書の内容に著作権法を侵害するコンテンツが含まれる可能性が高くなっている。文書作成者が著作権法を侵害してしまう原因としては、作成者が使用するコンテンツに著作権が主張されているかどうかを判定する手段がないこと、文書作成者がコンテンツを作成したコンテンツ作成者との交流が持てず著作権の許諾を申請する手段がないことなどが挙げられる。また、文書作成者またはコンテンツ作成者が閲覧者に対してコンテンツの著作権を主張したい場合に、文書作成者は閲覧者に対して著作権を主張する手段を設けることが困難なことも問題点として挙げられる。
【０００５】
上記のような著作権の侵害問題があるために、著作権保有者があまりにも著作権を保護することにとらわれて、現状では文書作成者が他者の著作物をＷｅｂなどに掲載することが制限されている。例えば、著作権を主張したい写真画像などをＷｅｂ上に掲載しておくと、閲覧者に不正利用される可能性が高くなるので、安易に写真画像などをＷｅｂ上に掲載できないなどの問題点が挙げられる。その反面、ナップスターなどに代表されるピアツーピアのファイル交換形式では、コンテンツの著作権保護方法が確立されていないため、著作権侵害の問題が社会問題にまで発展した。著作権侵害問題の他にも、作成者が公序良俗に反するコンテンツを電子文書内に挿入した場合に、閲覧者側では公序良俗に反するコンテンツを表示させないようにすることができないという問題点も挙げられる。
【０００６】
このような問題点を解決するために、従来でも、階層信用ディジタル署名システムによ

ってセキュリティレベルを設定したり、各ファイルごとにセキュリティレベルを設定したりしてセキュリティ精度を高めるようにしたりする、さまざまなセキュリティシステムが提案されている（例えば、特許文献１～５を参照。）。

【０００７】

【特許文献１】特開平６－１０３０５８号公報
【特許文献２】特開平１０－１０５４４９号公報
【特許文献３】特開２００１－１４３００９号公報
【特許文献４】特開２００１－３０９１５７号公報
【特許文献５】特開２００２－３２２８５号公報

【発明の開示】

【発明が解決しようとする課題】

【０００８】

しかしながら、上記各特許文献に開示されたセキュリティシステムをはじめとして、従来のセキュリティシステムでは、電子文書の表現力・編集能力・経時可読性・ポータビリティ性を向上させるに伴って電子文書の危険性が高まるため、閲覧者が文書作成者を信頼できない場合には、コンピュータ環境にとらわれずに豊かな表現力を持つ電子文書を安心して閲覧できなくなるという問題が生じる。また、従来より、作成者から閲覧者まで電子文書を送信する間に害意を持つ第三者によって電子文書が改ざんされることを防ぐために電子署名技術を適用する方法は周知であったが、この方法では文書作成者が意図的に障害を発生させようとしている場合に対するセキュリティ機能が考慮されていないため、閲覧者は電子文書を安心して閲覧することができなかった。

【０００９】

本発明は、上述した従来技術による問題点を解消するため、文書作成アプリケーションに電子文書の安全性を保証させることによって、電子文書の作成者を確認することなくても、有害な電子文書の閲覧が防止でき、電子文書閲覧の安全性が図れる電子文書ファイル、およびこの電子文書ファイルを作成する装置を提供することを目的とする。

【課題を解決するための手段】

【００１０】

上述した課題を解決し、目的を達成するため、請求項１の発明にかかる電子文書ファイルは、動作プログラム提供者（アプリ提供者）によって提供される、複数の動作プログラムおよび動作プログラム利用情報と、前記動作プログラム提供者または文書作成者によって作成されるコンテンツファイルおよびコンテンツファイル利用情報と、を含み構成されていることを特徴とする。

【００１１】

この請求項１に記載の発明によれば、電子文書ファイルの安全性をアプリ提供者が保証することによって、文書閲覧者にとって未知の文書作成者が作成した電子文書ファイルであっても安心して閲覧することができる。

【００１２】

また、請求項２にかかる電子文書ファイルは、請求項１に記載の発明において、前記複数の動作プログラムのうちの少なくとも一つは動作検証プログラムであり、該動作検証プログラムは、前記動作プログラム利用情報に基づき他の動作プログラムを動作させることを特徴とする。

【００１３】

この請求項２に記載の発明によれば、電子文書ファイルの安全性をアプリ提供者が保証することによって、文書閲覧者にとって未知の文書作成者が作成した電子文書ファイルであっても安心して閲覧することができる。

【００１４】

また、請求項３にかかる電子文書ファイルは、請求項１または２に記載の発明において、前記複数の動作プログラムのうち少なくとも一つはコンテンツ検証プログラムであり、該コンテンツ検証プログラムは、前記コンテンツファイル利用情報に基づきコンテンツフ

ファイルを利用することを特徴とする。

【００１５】

この請求項３に記載の発明によれば、電子文書ファイルの安全性をアプリ提供者が保証することによって、文書閲覧者にとって未知の文書作成者が作成した電子文書ファイルであっても安心して閲覧することができる。

【００１６】

また、請求項４にかかる電子文書ファイルは、請求項２または３に記載の発明において、前記動作検証プログラムは前記動作プログラム利用情報に基づき他の動作プログラムを動作させ、該動作プログラムのうちの少なくとも一つはコンテンツ検証プログラムであり、該コンテンツ検証プログラムは前記コンテンツファイル利用情報に基づきコンテンツファイルの利用を制限することを特徴とする。

【００１７】

この請求項４に記載の発明によれば、電子文書ファイルの安全性をアプリ提供者が保証することによって、文書閲覧者にとって未知の文書作成者が作成した電子文書ファイルであっても安心して閲覧することができる。

【００１８】

また、請求項５にかかる電子文書ファイルは、請求項２～４のいずれか一つに記載の発明において、前記動作検証プログラムは、前記動作プログラムおよび前記動作プログラム利用情報の改ざんを検証し、または前記動作プログラムの動作を制限することを特徴とする。

【００１９】

この請求項５に記載の発明によれば、電子文書ファイルの安全性をアプリ提供者が保証することによって、文書閲覧者にとって未知の文書作成者が作成した電子文書ファイルであっても安心して閲覧することができる。

【００２０】

また、請求項６にかかる電子文書ファイルは、請求項３～５のいずれか一つに記載の発明において、前記コンテンツ検証プログラムは、前記コンテンツファイルおよび前記コンテンツファイル利用情報の改ざんを検証し、または前記コンテンツファイルの利用を制限することを特徴とする。

【００２１】

この請求項６に記載の発明によれば、コンテンツファイル利用情報にコンテンツファイルの復号化情報および使用制限情報を保存することによって、文書作成者が作成したコンテンツファイルの改ざんの検証や、コンテンツファイル内に含まれる著作権を侵害するコンテンツや公序良俗に反するコンテンツの使用を禁止することができる。

【００２２】

また、請求項７にかかる電子文書ファイルは、請求項２～６のいずれか一つに記載の発明において、前記動作検証プログラムによって前記動作プログラムの安全性が保証されない場合には、当該電子文書ファイルの使用者に前記動作プログラムの安全性が保証されていないことを認識させることを特徴とする。

【００２３】

この請求項７に記載の発明によれば、電子文書に危険性がある場合に文書閲覧者に電子文書の安全性が保証されていないことを明示し、文書閲覧者に対して有害な電子文書ファイルを閲覧しないように示唆することができる。

【００２４】

また、請求項８にかかる電子文書ファイルは、請求項１～７のいずれか一つに記載の発明において、前記動作プログラム、前記動作プログラム利用情報、前記コンテンツファイル、および前記コンテンツファイル利用情報は、単一ファイルにカプセル化されていることを特徴とする。

【００２５】

この請求項８に記載の発明によれば、電子文書ファイルの安全性を保ちつつ、コンピュ

ータデバイスおよびネットワーク環境に依存しない電子文書の相互交換・伝送・表示・編集・保存・印刷処理などを実現し、電子文書の経時可読性を半永久的に保存する電子文書ファイルを提供できる。

【００２６】

また、請求項９にかかる電子文書ファイル作成装置は、複数の動作プログラムおよび動作プログラム利用情報を読み込む読込手段と、コンテンツファイルおよびコンテンツファイル利用情報を作成するコンテンツファイル作成手段と、請求項１～８のいずれか一つに記載の電子文書ファイルに格納する各情報を作成する情報作成手段と、を含み構成されていることを特徴とする。

【００２７】

この請求項９に記載の発明によれば、請求項１～８に記載された電子文書ファイルを作成する電子文書ファイル作成装置を提供できる。

【００２８】

また、請求項１０にかかる電子文書ファイル作成装置は、請求項９に記載の発明において、前記コンテンツファイル作成手段は、請求項１～８のいずれか一つに記載の電子文書ファイル内に格納されている複数の動作プログラムの一つを動作させることを特徴とする。

【００２９】

この請求項１０に記載の発明によれば、請求項１～８に記載された電子文書ファイルを作成する電子文書ファイル作成装置を提供できる。

【００３０】

また、請求項１１にかかる電子文書ファイル作成装置は、請求項９または１０に記載の発明において、さらに前記コンテンツファイル作成手段により作成されたコンテンツファイルおよびコンテンツファイル利用情報に基づいて、動作プログラム提供者によって提供される複数の動作プログラムおよび動作プログラム利用情報を付加する動作プログラム付加手段を備えていることを特徴とする。

【００３１】

この請求項１１に記載の発明によれば、文書作成者が作成したコンテンツファイルに応じて動作プログラムを付加することが可能な電子文書ファイル作成装置を提供できる。

【発明の効果】

【００３２】

本発明にかかる電子文書ファイル、電子文書ファイル作成装置によれば、文書作成アプリケーションに電子文書の安全性を保証させることによって、電子文書の作成者を確認することなくても、有害な電子文書の閲覧が防止でき、電子文書閲覧の安全性を図ることができるという効果を奏する。

【発明を実施するための最良の形態】

【００３３】

以下に添付図面を参照して、この発明にかかる電子文書ファイル、電子文書ファイル作成装置の好適な実施の形態を詳細に説明する。

【００３４】

（実施の形態１）

図１は、実施の形態１にかかる電子文書ファイルの構造を示す図である。この電子文書ファイルは、コンテンツファイル１０１，１０２、コンテンツファイル利用情報１０３、動作プログラム１０４，１０５，１０６、および動作プログラム利用情報１０７を含み構成される。コンテンツファイル１０１，１０２は、文書作成者が作成する文書内容（コンテンツ）を保存するファイルである。コンテンツファイル１０１，１０２には、文書作成者が文書内容を記述しやすいように予めテンプレート情報を挿入しておいてもよい。テンプレート情報の例としては、コンテンツファイル１０１，１０２に対する動作処理をグラフィックユーザインターフェース形式で実行させるために必要となるアイコン画像や、コンテンツファイル１０１，１０２を作成した際の背景画像などがある。

【００３５】
また、コンテンツファイル利用情報１０３は、コンテンツファイル１０１，１０２内に保存されたコンテンツ情報に対する利用情報（コンテンツに対する使用制限や使用方法、コンテンツファイル１０１，１０２の特徴量など）を含んでいる。コンテンツファイル１０１，１０２内にあるコンテンツ情報の改ざん検証や使用制限などが必要ないときは、コンテンツファイル利用情報１０３は不要である。

【００３６】
動作プログラム１０４，１０５，１０６は、アプリ提供者（動作プログラム提供者）から提供されるプログラムであり、コンテンツファイル１０１，１０２に対する相互交換・伝送・表示・編集・保存・印刷処理などを実行するプログラムや動作プログラム１０４，１０５，１０６およびコンテンツファイル１０１，１０２を検証するプログラムなどを含んでいる。文書作成者は、動作プログラム１０４，１０５，１０６に含まれる文書作成プログラムを使用してコンテンツファイル１０１，１０２に文書内容を記述する。具体的な動作プログラム１０４，１０５，１０６の機能としては、ワードプロセッサ機能・画像編集機能・音楽編集機能・動画編集機能などが挙げられる。文書閲覧者は、動作プログラム１０４，１０５，１０６に含まれる文書閲覧プログラムを使用してコンテンツファイルに記述された文書内容を閲覧する。

【００３７】
動作プログラム利用情報１０７は、動作プログラム１０４，１０５，１０６に対する使用制限や使用方法、動作プログラム１０４，１０５，１０６の改ざん検証をする際に利用する動作プログラムの特徴量や復号化情報などを含んでいる。動作プログラム利用情報１０７に含まれる情報等をＸＭＬ形式で記載したフォーマットを図２に示す。このフォーマット２０１の特徴量情報タグ内の属性情報として記載されているアルゴリズムは、動作プログラムの特徴量を算出する手法を示す。このフォーマット２０１では、後述するＳＨＡという手法で特徴量を算出している。また公開鍵情報は、暗号化された動作プログラムを復号化するための情報である。動作プログラムタグ内の属性情報として記載された位置情報はカプセル化文書内に動作プログラムが保存されている位置を示す。また特徴量には、動作プログラムの特徴量を算出した結果が記載されている。ここで、動作プログラムを検証する際に特徴量情報を使用しない場合には、特徴量やアルゴリズムなどの属性情報を省略してもよい。

【００３８】
図１に示した構造の電子文書ファイルに対する安全性の保証は、動作プログラムの一つである動作検証プログラム１０６が、動作プログラム利用情報１０７に基づいて動作プログラムの改ざん検証を実行することにより実現する。動作検証プログラム１０６の実行例としては、電子文書ファイル内にある暗号化された動作プログラムを復号化し、動作プログラムの署名情報を検証することによって行う。動作プログラムの暗号化手法の例としては、共通鍵方式を用いる手法や公開鍵方式を用いる手法など挙げられる。以下では、公開鍵方式を用いた例を述べる。公開鍵方式では、アプリ提供者が秘密鍵を生成し、この秘密鍵によって動作プログラムを暗号化する。秘密鍵と対となる公開鍵は文書作成者と文書閲覧者が取得できるように配布する。秘密鍵を利用した暗号化には、ＲＳＡやＤＳＡ（Digital Signature Algorithm）などを利用する。ＲＳＡの暗号アルゴリズムは大きな数値の素因数分解を実行することが困難であるという事実に依存したものであり、２０００ビット以上の係数を持つ鍵であれば安全であると一般的に知られている。ＲＳＡ、ＤＳＡなどは、様々な企業から提供されているため、簡単に使用することが可能である。

【００３９】
次に、この実施の形態１の電子文書ファイルを用いた電子文書認証の概略を図３に基づいて説明する。この電子文書認証では、アプリ提供者・文書作成者・文書閲覧者が存在し、アプリ提供者は電子文書ファイルを生成して文書作成者へ配布する（ステップＳ３０１）。文書作成者は配布された電子文書ファイルを使用して文書内容をコンテンツファイルに記述し、文書閲覧者へ配布する（ステップＳ３０２）。文書閲覧者は、配布された電子

文書ファイルの改ざん検証を行った後に動作プログラムを実行させ、改ざんされていなければコンテンツファイルを閲覧させる（ステップＳ３０３）。

【００４０】

以下、この電子文書認証を詳細に説明する。まず、アプリ提供者が実行する処理について説明する。図４は、アプリ提供者による電子文書認証のための処理手順を示すフローチャートである。まず、アプリ提供者は、秘密鍵・公開鍵を生成する（ステップＳ４０１）。次に、複数の動作プログラム・コンテンツファイルを生成する（ステップＳ４０２）。次いで、アプリ提供者が生成した秘密鍵で動作プログラムを暗号化する（ステップＳ４０３）。動作プログラム利用情報を生成して、公開鍵情報を保存する（ステップＳ４０４）。必要に応じて、コンテンツファイル利用情報を生成する（ステップＳ４０５）。暗号化された動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から構成される電子文書ファイルを作成する（ステップＳ４０６）。この後、作成した電子文書ファイルを文書作成者へ配布する。

【００４１】

次に、文書作成者が実行する処理について説明する。図５は、文書作成者による電子文書認証のための処理手順を示すフローチャートである。まず、文書作成者はアプリ提供者から配布された電子文書ファイルを取得し、電子文書ファイル内にある複数の動作プログラムの利用情報に保存された公開鍵情報で暗号化された動作プログラムの一つである文書作成プログラムを復号化する（ステップＳ５０１）。次に、動作プログラムの署名情報に基づき認証を行い、改ざんの有無を検証する（ステップＳ５０２）。改ざんがある場合（ステップＳ５０２：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ５０３）、そのまま処理を終了する（すなわち、文書作成を中止する）。改ざんがない場合（ステップＳ５０２：Ｎｏ）は、文書作成プログラムを使用して閲覧者へ伝えたい文書内容を作成し、コンテンツファイルに保存する（ステップＳ５０４）。

【００４２】

また、必要に応じて、コンテンツファイル内にあるコンテンツに対する使用制限などをコンテンツファイル利用情報に記述する（ステップＳ５０５）。このようにして文書作成者によって文書内容が記載された電子文書ファイルを文書閲覧者に配布する。なお、配布する電子文書ファイル内の動作プログラムの一つである文書作成プログラムは、配布した際または文書閲覧者が取得した際に自動的に消去してもよい。文書作成プログラムの消去は、アプリ提供者が予め文書作成プログラム内に文書作成プログラムデータの消去プログラムを実装しておくことによって行うことができる。

【００４３】

続いて、文書閲覧者が実行する処理について説明する。図６は、文書閲覧者による電子文書認証のための処理手順を示すフローチャートである。まず、文書閲覧者は、文書作成者から配布された電子文書ファイルを取得すると、電子文書ファイル内にある動作プログラム利用情報に保存された公開鍵情報により、動作プログラムを復号化する（ステップＳ６０１）。次に、動作プログラムの一つである動作検証プログラムを使用して、アプリ提供者が動作プログラムに署名した署名情報を認証し、害意を持つ文書作成者による動作プログラムの改ざん検証を行う（ステップＳ６０２）。改ざんがある場合（ステップＳ６０２：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ６０３）、そのまま処理を終了する（すなわち、文書閲覧を中止する）。

【００４４】

改ざんがない場合（ステップＳ６０２：Ｎｏ）は、コンテンツファイルを表示させるため動作プログラムの一つである文書閲覧プログラムを実行させ、コンテンツファイルを表示させる（ステップＳ６０４）。改ざんされた動作プログラムを実行する恐れがなくなるので、文書閲覧者はアプリ提供者から提供された動作プログラムだけを実行させて電子文書ファイルを閲覧することができる。以上の工程を経ることにより、文書閲覧者は文書作成者が文書に有害情報等を付加した場合であっても、その悪影響を受けることを回避できる。

【００４５】
この実施の形態１における電子文書ファイルは、動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報が分割された状態で配布されても、単一ファイルにカプセル化された状態で配布されてもよい。ただし、分割された状態で電子文書ファイルを配布する場合には、動作プログラムと動作プログラム利用情報、コンテンツファイルとコンテンツファイル利用情報、動作プログラムとコンテンツファイルとの対応が明白になるようにリンク情報を追加するなどの工夫が必要となる。単一ファイルにカプセル化することによって、取り扱いの便利な電子文書ファイルを提供することができる。ただし、前述した動作プログラム利用情報として公開鍵情報を保存する場合には、害意を持つ文書作成者によって、動作プログラム利用情報に保存された公開鍵情報と動作プログラムとの両方が改ざんされてしまい、文書閲覧者が改ざんされた公開鍵情報で動作プログラムの改ざん検証を実行する危険性がある。

【００４６】
したがって、公開鍵情報が改ざんされないようにすることが重要である。この解決法の一つとして、公開鍵情報を電子文書ファイルと異なる経路で文書閲覧者に配布することが挙げられる。公開鍵情報の配布形態としては、アプリ提供者が所有するホームページ上に掲載して自由にダウンロードできてもよいし、ＣＤ－ＲＯＭやＤＶＤ－ＲＯＭなどの記憶媒体に公開鍵情報を記録して文書閲覧者へ配布してもよい。その他の解決法としては、文書閲覧者側のコンピュータに予め保持されている第三認証局の公開鍵情報を利用して、電子文書ファイル内の公開鍵情報の改ざん検証を行うことが挙げられる。公開鍵情報の一例としては、規格化されている認証書フォーマット（いわゆるＸ５０９フォーマット）などを使用してもよい。

【００４７】
また、暗号化・復号化させるデータ量を減少させて動作プログラムの実行を高速化するため、動作プログラムの特徴量を算出し、算出した特徴量を暗号化・復号化する方式にしてよい。特徴量を使用する場合には、動作プログラム利用情報にアプリ提供者の秘密鍵によって暗号化した動作プログラムの特徴量を保存する。ここで動作プログラムの特徴量とは、動作プログラムに対する電子指紋のことである。この特徴量の算出方法としては、ＳＨＡ１（Secure-Hash-Algorithm）などが挙げられる。ＳＨＡ１の基本的な特性としては、動作プログラムを１ビットでも変更すると動作プログラムの特徴量も変更されることと、特徴量の偽造を試みても元の特徴量と同じ特徴量を持つ偽造動作プログラムを生成できないことの２点が挙げられる。この特性により、動作プログラムと動作プログラムの特徴量が一対一の関係となることが立証される。

【００４８】
以下、この場合の電子文書ファイルの認証について簡単に説明する。まず、アプリ提供者は、秘密鍵で動作プログラムの特徴量だけを暗号化して文書作成者へ配布する。文書作成者は、動作プログラム内にある文書作成プログラムを使用しコンテンツファイルに文書内容を記述して文書閲覧者へ配布する。文書閲覧者は、公開鍵情報で暗号化された動作プログラムの特徴量を復号化した結果と動作プログラムの特徴量を算出した結果を比較することによって動作プログラムの改ざん検証を実行する。ここで、文書作成者によって動作プログラムの改ざんがなく、同一なアルゴリズムを適用して同一な動作プログラムの特徴量が算出されれば、アプリ提供者側から配布された暗号化された動作プログラムの特徴量を復号化した値と閲覧者側で算出した動作プログラムの特徴量の値は一致する。特徴量算出方法に関しては公開されているので簡単に同一なアルゴリズムを適用して特徴量を算出することができる。以上より２つの特徴量を比較することによって動作プログラムの改ざんが検証できる。

【００４９】
また、動作プログラム利用情報に複数の動作プログラムの安全性に応じて動作プログラムの実行を制限させる動作プログラム制限情報を記録させてもよい。動作プログラム制限情報として動作プログラムを暗号化する鍵を複数用意し、動作プログラムの動作権限に応

じて、暗号化・復号化する鍵を変えることによって実現できる。一例としては、図７に示す表のように、ファイルの読み書き、ネットワークの送受信等を許可、非許可の動作権限のモードに応じて暗号化する鍵を用意し、動作プログラムを復号化する際に使用した鍵の動作権限を文書閲覧者に表示することで文書閲覧者に許可をもらって、そのモードの動作権限で動作プログラムを起動してもよい。

【００５０】

また、前述の複数の動作プログラムは中間言語コードで記述されていることが好ましい。プログラムが中間言語で記述されていれば、この中間言語を解釈実行できるコンパイラまたはインタプリタプログラムがコンピュータにインストールされている状況においてコンピュータの機種依存性がなくなる。このような中間言語としては、例えばｊａｖａ言語がある。ｊａｖａの技術を利用することにより動作プログラムの特徴量算出処理なども簡易になる。しかしながら、現在のｊａｖａ言語はコンピュータ上で動作するアプリケーションを開発するものであって、この実施の形態１に示したような電子文書ファイルを定義するものではない。

【００５１】

次に、この実施の形態１の電子文書ファイルを用いた電子文書認証を行うためのシステム構成について説明する。図８は、この実施の形態１にかかる電子文書ファイルを用いて行う電子文書認証システムの全体構成図である。この電子文書認証システムは、アプリ提供者８０１、文書作成者８０２、および文書閲覧者８０３により構成される。図中の矢印は、電子文書ファイルの送信方向を示している。

【００５２】

また、この電子文書認証システムは、図９のように構成することも可能である。図９に示した電子文書認証システムでは、初めに文書作成者９０２がアプリ提供者９０１によって定められたコンテンツファイル・コンテンツファイル利用情報のフォーマットに従って、コンテンツファイル・コンテンツファイル利用情報を作成する。作成したコンテンツファイル・コンテンツファイル利用情報をアプリ提供者９０１へ送信する。この際、電子文書ファイルを配布したい文書閲覧者９０３へのアドレス情報もアプリ提供者９０１へ送信してもよい。アプリ提供者９０１は送信されたコンテンツファイル内に記述されたコンテンツ（静止画、動画、三次元画像、音楽など）を動作させるために必要な動作プログラム・動作プログラム利用情報を選択する。選択された動作プログラム・動作プログラム利用情報と文書作成者９０２によって作成されたコンテンツファイル・コンテンツファイル利用情報から図１に示した構造の電子文書ファイルを作成する。

【００５３】

作成した電子文書ファイルを文書作成者９０２から送信された文書閲覧者９０３のアドレス情報に従って文書閲覧者９０３へ配布する。文書閲覧者９０３は、電子文書ファイル内にある動作プログラムの改ざん検証を行い、改ざんがなければ動作プログラムを実行してコンテンツファイルを閲覧する。この例では、アプリ提供者９０１が電子文書ファイルを文書閲覧者へ配布するようにしたが、アプリ提供者９０１は作成した電子文書ファイルを一旦文書作成者９０２へ返信し、文書作成者９０２が文書閲覧者９０３へ電子文書ファイルを配布するようにしてもよい。

【００５４】

次に、電子文書認証システム内でアプリ提供者・文書作成者・文書閲覧者が使用する代表的なハードウェア構成を図１０に示す。このハードウェアは、電子文書認証システムにおける各種の制御および処理を行うＣＰＵ（central processing unit）１００１、ＲＡＭ（random access memory）１００２、ＨＤＤ（hard disk drive）１００３、マウス等のポインティングデバイス，キーボード，ボタン等の入力インターフェース（以下Ｉ／Ｆという）１００４、表示Ｉ／Ｆ１００５を介して接続されるＣＲＴ（cathode ray tube）等のディスプレイ１００６、ＣＤ－ＲＷ（compact disk re－write－able）ドライブ等の記録装置１００７、および画像入力部やプリンタ等の外部機器やインターネット等の電気通信回線と有線または無線接続するための外部Ｉ／Ｆ１００８が、バス１００９を介

して接続されて構成される。

【００５５】

ＲＡＭ１００２は、ＣＰＵ１００１の作業領域として利用されるとともに、電子文書認証システム内の各工程を実行するための複数の動作プログラムや、その他制御プログラムなどの固定情報の記録領域として利用される。動作プログラムは、例えば記録装置１００７を介してＲＡＭ１００２にロードされ、またはＨＤＤ１００３に一旦保存された後に必要なときにＲＡＭ１００２にロードされ、または外部Ｉ／Ｆ１００８に接続された電気通信回線を介してＲＡＭ１００２にロードされる。

【００５６】

文書作成者は、図１０に示したようなハードウェアを用いて図１に示した構造の電子文書ファイルを読み込んで、コンテンツファイルに文書閲覧者へ伝えたい文書内容を記述する。この際、記録装置１００７には、電子文書ファイル内にある動作プログラムと動作プログラム利用情報を読み込むプログラムと、文書作成プログラムと、読み込んだ動作プログラム・動作プログラム利用情報と作成したコンテンツファイル・コンテンツファイル利用情報から電子文書ファイルを作成する電子文書ファイル作成プログラムの３つのプログラムが記録されている。

【００５７】

（実施の形態２）

この実施の形態２では、複数の動作プログラムの一つにコンテンツ検証プログラムを追加することによって、文書作成者が作成したコンテンツファイルの改ざん検証やコンテンツファイル内に著作権を侵害するコンテンツや公序良俗に反するコンテンツを使用することができなくする電子文書ファイルの例を示す。具体的な電子文書ファイルの構造は、図１に示したものと同様である。コンテンツ検証プログラムはコンテンツファイル利用情報に基づいてコンテンツファイルの改ざん検証・コンテンツファイル内のコンテンツ使用規制を行う。

【００５８】

以下、この実施の形態２のコンテンツファイルの改ざん検証について説明する。まず、アプリ提供者が実行する処理を示す。図１１は、アプリ提供者によるコンテンツファイルの改ざん検証のための処理手順を示すフローチャートである。まず、アプリ提供者は、秘密鍵・公開鍵を生成する（ステップＳ１１０１）。次に、動作プログラム・コンテンツファイルを生成する（ステップＳ１１０２）。次いで、アプリ提供者の秘密鍵で動作プログラムを暗号化する（ステップＳ１１０３）。動作プログラム利用情報を生成して、公開鍵情報を保存する（ステップＳ１１０４）。必要に応じて、コンテンツファイル利用情報を生成する（ステップＳ１１０５）。暗号化された動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から構成される電子文書ファイルを作成する（ステップＳ１１０６）。この後、作成した電子文書ファイルを文書作成者へ配布する。

【００５９】

次に、文書作成者が実行する処理について説明する。図１２は、文書作成者によるコンテンツファイルの改ざん検証のための処理手順を示すフローチャートである。まず、文書作成者はアプリ提供者から配布された電子文書ファイルを取得し、電子文書ファイル内の動作プログラム利用情報に保存された公開鍵情報で暗号化された動作プログラムの復号化を行う（ステップＳ１２０１）。次に、動作プログラムの署名情報に基づき認証を行い、改ざんの有無を検証する（ステップＳ１２０２）。改ざんがある場合（ステップＳ１２０２：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ１２０３）、そのまま処理を終了する（すなわち、文書作成を中止する）。改ざんがない場合（ステップＳ１２０２：Ｎｏ）は、アプリ提供者と同様に文書作成者特有の秘密鍵・公開鍵を生成する（ステップＳ２１０４）。動作プログラム内の文書作成プログラムを使用してコンテンツファイルに閲覧者へ伝えたい文書内容を記述する（ステップＳ１２０５）。そして、コンテンツファイルからコンテンツファイルの特徴量を実施の形態１の場合と同様にして算出する（ステ

ップＳ１２０６）。算出した特徴量を文書作成者が保持する秘密鍵で暗号化し、コンテンツファイル利用情報として暗号化したコンテンツファイルの特徴量を保存する（ステップＳ１２０７）。この後、アプリ提供者から提供された動作プログラム・動作プログラム利用情報と、文書作成者が作成したコンテンツファイル・コンテンツファイル利用情報に基づいて作成された電子文書ファイルが文書閲覧者へ配布される。

【００６０】

続いて、文書閲覧者が実行する処理について説明する。図１３は、文書閲覧者によるコンテンツファイルの改ざん検証のための処理手順を示すフローチャートである。まず、文書閲覧者は、文書作成者から配布された電子文書ファイルを取得すると、電子文書ファイル内にある動作プログラム利用情報に保存された公開鍵情報により、動作プログラムを復号化する（ステップＳ１３０１）。次に、動作プログラムの一つである動作検証プログラムを使用して、アプリ提供者が動作プログラムに署名した署名情報を認証し、動作プログラムの改ざん検証を行う（ステップＳ１３０２）。改ざんがある場合（ステップＳ１３０２：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ１３０３）、そのまま処理を終了する（すなわち、文書閲覧を中止する）。改ざんがない場合（ステップＳ１３０２：Ｎｏ）は、動作プログラムの一つであるコンテンツ検証プログラムを実行し、コンテンツファイル利用情報に保存されている暗号化されたコンテンツファイルの特徴量を復号化する。

【００６１】

そして、復号化した結果と新たにコンテンツファイルから特徴量を算出した結果を比較してコンテンツファイルの改ざん検証を行う（ステップＳ１３０４）。改ざんがある場合（ステップＳ１３０４：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ１３０３）、そのまま処理を終了する（すなわち、文書閲覧を中止する）。改ざんがない場合（ステップＳ１３０４：Ｎｏ）は、コンテンツファイルを表示させるため動作プログラムの一つである文書閲覧プログラムを実行させ、コンテンツファイルを表示させる（ステップＳ１３０５）。以上のような工程を経ることによって、文書閲覧者は、配布された電子文書ファイル内にある文書内容が、コンテンツファイルに署名した文書作成者によって作成されたことを確認できる。

【００６２】

また、コンテンツファイル内のコンテンツに対する著作権を規制する場合には、コンテンツファイル利用情報にコンテンツの使用規制情報を保存する。コンテンツの著作権侵害は、文書作成者が電子文書ファイルを作成するときにアプリ提供者やコンテンツ提供者が提供するコンテンツを無断で使用すること、文書閲覧者が電子文書ファイルにあるコンテンツを無断で使用することによって生じる。以下、コンテンツ著作権を保護するために行う処理について説明する。

【００６３】

まず、アプリ提供者が実行する処理について説明する。図１４は、アプリ提供者によるコンテンツ著作権保護のための処理手順を示すフローチャートである。まず、アプリ提供者は、秘密鍵・公開鍵を生成する（ステップＳ１４０１）。次に、動作プログラム・コンテンツファイルを生成する（ステップＳ１４０２）。次いで、アプリ提供者の秘密鍵で複数の動作プログラムを暗号化する（ステップＳ１４０３）。動作プログラム利用情報を生成して、公開鍵情報を保存する（ステップＳ１４０４）。文書作成者に提供するコンテンツをコンテンツファイル内に挿入する（ステップＳ１４０５）。コンテンツファイル内のコンテンツ（著作権を主張したいコンテンツ）を暗号化する（ステップＳ１４０６）。暗号化されたコンテンツを復号化するための復号化情報をコンテンツファイル利用情報に保存する（ステップＳ１４０７）。

【００６４】

ここでコンテンツファイル利用情報に保存する復号化情報を調整することによって、文書作成者が使用できるコンテンツに制限を与えることができる。例えば、コンテンツ情報に音楽情報が保存されているときには、コンテンツファイル利用情報内にある復号化情報

に応じて、音楽情報の音質を変化させて復号化してもよい。そして、暗号化された動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から構成される電子文書ファイルを作成する（ステップＳ１４０８）。この後、作成した電子文書ファイルを文書作成者へ配布する。

【００６５】

次に、文書作成者が実行する処理について説明する。図１５は、文書作成者によるコンテンツ著作権保護のための処理手順を示すフローチャートである。まず、文書作成者はアプリ提供者から配布された電子文書ファイルを取得し、電子文書ファイル内の動作プログラム利用情報に保存された公開鍵情報で暗号化された動作プログラムの復号化を行う（ステップＳ１５０１）。次に、動作プログラムの署名情報に基づき認証を行い、改ざんの有無を検証する（ステップＳ１５０２）。改ざんがある場合（ステップＳ１５０２：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ１５０３）、そのまま処理を終了する（すなわち、文書作成を中止する）。改ざんがない場合（ステップＳ１５０２：Ｎｏ）は、コンテンツファイル利用情報に保存された復号化情報に応じてコンテンツファイル内の暗号化されたコンテンツを復号化する（ステップＳ１５０４）。

【００６６】

なお、ここで動作プログラムの一つにコンテンツ復号化情報要求プログラムを追加し、コンテンツを復号化する際にコンテンツファイル利用情報内に保存された復号化情報が足りない場合には、アプリ提供者へ復号化情報を要求するプログラムを加えてもよい。そして、動作プログラムと復号化されたコンテンツを使用してコンテンツファイルに文書閲覧者へ伝えたい文書内容を記述する（ステップＳ１５０５）。この後、文書閲覧者に対して電子文書ファイルを配布する。

【００６７】

続いて、文書閲覧者が実行する処理について説明する。図１６は、文書閲覧者によるコンテンツ著作権保護のための処理手順を示すフローチャートである。まず、文書閲覧者は、文書作成者から配布された電子文書ファイルを取得すると、電子文書ファイル内にある動作プログラム利用情報に保存された公開鍵情報により、動作プログラムを復号化する（ステップＳ１６０１）。次に、動作プログラムの一つである動作検証プログラムを使用して、アプリ提供者が動作プログラムに署名した署名情報を認証し、害意を持つ文書作成者による動作プログラムの改ざん検証を行う（ステップＳ１６０２）。改ざんがある場合（ステップＳ１６０２：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ１６０３）、そのまま処理を終了する（すなわち、文書閲覧を中止する）。

【００６８】

改ざんがない場合（ステップＳ１６０２：Ｎｏ）は、コンテンツファイルを表示させるため動作プログラムの一つである文書閲覧プログラムを実行させ、コンテンツファイルを表示させる（ステップＳ１６０４）。このような工程を経ることによって、アプリ提供者によるコンテンツファイル内にあるコンテンツに利用制限を施し、コンテンツに対する著作権主張を行うことが容易になる。文書作成者が作成したコンテンツの著作権を保護するときは、文書作成者が作成したコンテンツに対して暗号化を行い、暗号化したコンテンツを復号化する復号化情報をコンテンツ利用情報に保存する。コンテンツ利用情報に保存する復号化情報を調整して、文書閲覧者が使用できるコンテンツに制限を与えることができる。

【００６９】

さらに、動作プログラムの一つに著作権の使用許可願い通知プログラムを付加することによって、文書作成者または文書閲覧者が電子文書ファイル内に著作権を侵害するコンテンツを利用した場合は、文書作成者または文書閲覧者にコンテンツ著作権の保持者または著作権団体を明示し、著作権保持者との交流を円滑にして著作権の使用許可願いを簡易に実行できるようにしてもよい。

【００７０】

図１７は、著作権の使用許可願い通知を行う場合の処理手順を示すフローチャートであ

る。まず、文書閲覧者側でコンテンツを使用する動作プログラムが実行されると自動的にコンテンツファイル利用情報に記載されているコンテンツ著作権保持者のアドレス情報を読み込む（ステップＳ１７０１）。次に、読み込んだアドレス先（コンテンツ著作権保持者）へ電子文書ファイル内のコンテンツの使用の許可を確認するためのメールを送信する（ステップＳ１７０２）。確認メールを受信したコンテンツ著作権保持者は、当該コンテンツの使用を許可するか否かの決定を行う（ステップＳ１７０３）。当該コンテンツの使用を許可する場合（ステップＳ１７０３：Ｙｅｓ）は、コンテンツファイル利用情報の使用許可情報を変更し、コンテンツファイル内のコンテンツを使用する動作プログラムを実行させる（ステップＳ１７０４）。当該コンテンツの使用を許可しない場合（ステップＳ１７０３：Ｎｏ）は、不正処理が行われたと判断し（ステップＳ１７０５）、そのまま処理を終了する。なお、使用許可願い通知プログラムはコンテンツの使用通知だけでなく、コンテンツ使用料の支払い処理を同時に行ってもよい。また、コンテンツの不正利用を防ぐため、電子文書ファイルを閲覧している文書閲覧者側のコンピュータのアドレスと文書閲覧者のユーザ名を、コンテンツ著作権保持者へ送信してもよい。

【００７１】

図１８は、この使用許可願い通知を行う場合に表示される、コンテンツファイル利用情報のフォーマット１８０１，１８０２を示す図である。なお、フォーマット１８０１，１８０２の形式は、動作プログラム利用情報と同様にＸＭＬ形式で記載してもよい。

【００７２】

また、文書作成者がコンテンツファイルへ外部コンテンツを挿入するときには、アプリ提供者が提供する文書作成プログラムを介さないと挿入できないようにすることによって、コンテンツファイルへの外部コンテンツの挿入を規制してもよい。文書作成プログラムでは、挿入する外部コンテンツのヘッダ情報を読み込み、コンテンツの中に過激な暴力シーンなどの公序良俗に反するコンテンツが含まれることが記述されている場合には、コンテンツファイルにコンテンツを挿入することを制限する。また、文書作成者が外部コンテンツをコンテンツファイルに挿入したいときには、挿入したい外部コンテンツをアプリ提供者へ送信し、アプリ提供者が外部コンテンツに対するコンテンツファイル利用情報を生成して、コンテンツファイル内に外部コンテンツを挿入するようにしてもよい。コンテンツの利用制限は、文書閲覧者側で実行してもよい。文書閲覧プログラムに文書閲覧者の個人情報を登録し、文書閲覧者の好みによってコンテンツファイル内にあるコンテンツの表示制限を実行してもよい。

【００７３】

（実施の形態３）

電子文書認証システムでは、動作プログラムが安全であることを保証する電子文書ファイルと保証しない電子文書ファイルの２種類のファイルが、アプリ提供者から文書作成者および文書閲覧者に提供される。未知の文書作成者から配布された電子文書ファイルが、安全性が保証されていない電子文書ファイルであると、閲覧者は安心して電子文書ファイル内にあるコンテンツファイルを閲覧できなくなる。安全性が保証されている電子文書ファイルは、動作プログラムがアプリ提供者によって暗号化されているが、安全性が保証されていない電子文書ファイルは、動作プログラムがアプリ提供者によって暗号化されていない。このため、電子文書ファイル内にある動作プログラムの改ざんの検証を行うことができない。そこで、かかる不都合を解決するのがこの実施の形態３である。

【００７４】

図１９は、実施の形態３にかかる電子文書ファイルを用いて行う電子文書認証システムの全体構成図である。この電子文書認証システムは、アプリ提供者１９０１、文書作成者１９０２、および文書閲覧者１９０３により構成される。図中の矢印は、電子文書ファイルの送信方向を示している。

【００７５】

この実施の形態３にかかる電子文書ファイルを用いて行う電子文書認証の概略を図２０に基づいて説明する。この電子文書認証では、アプリ提供者・文書作成者・文書閲覧者が

存在し、アプリ提供者は電子文書ファイル（安全性の保証はない）を生成して文書作成者へ配布する（ステップＳ２００１）。文書作成者は、動作プログラムの一つである文書作成プログラムを使用して、文書閲覧者に伝えたい文書内容をコンテンツファイルに作成し、電子文書ファイルをアプリ提供者へ返信する（ステップＳ２００２）。文書作成者から受け取った電子ファイル内の動作プログラムおよび動作プログラム利用者情報を削除し、代わりに秘密鍵で暗号化した動作プログラムおよび動作プログラム利用者情報を新たに作成して電子文書ファイルに挿入し、文書作成者へ送信する（ステップＳ２００３）。アプリ提供者から取得した電子文書ファイルのコンテンツファイルに、文書閲覧者に伝えたい内容を記述し、文書閲覧者へ配布する（ステップＳ２００４）。文書閲覧者は、電子文書ファイルと公開鍵を取得し、暗号化された動作プログラムを公開鍵で復号化し、動作プログラムの改ざんを検証し、改ざんされていなければそのプログラムを実行させてコンテンツファイルを閲覧する（ステップＳ２００５）。

【００７６】

以下、この電子文書認証を詳細に説明する。まず、アプリ提供者が実行する第１の処理について説明する。図２１は、アプリ提供者による電子文書認証のための第１の処理手順を示すフローチャートである。まず、アプリ提供者は、動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報を生成する（ステップＳ２１０１）。そして、ステップＳ２１０１で生成された動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から電子文書ファイルを作成する（ステップＳ２１０２）。この後、作成した電子文書ファイルを文書作成者へ配布する。

【００７７】

次に、文書作成者が実行する第１の処理について説明する。図２２は、文書作成者による電子文書認証のための第１の処理手順を示すフローチャートである。まず、文書作成者は、アプリ提供者から配布された電子文書ファイルを取得し、動作プログラムの一つである文書作成プログラムを使用して文書閲覧者へ伝えたい文書内容をコンテンツファイル内に記述する（ステップＳ２２０１）。この際、実施の形態１の場合とは異なり、動作プログラムは暗号化されていないため、文書作成者により改ざんされるおそれがある。次に、必要に応じて、コンテンツファイル内にあるコンテンツに対する使用制限などをコンテンツファイル利用情報に記述する（ステップＳ２２０２）。次に、文書作成者は、文書閲覧者から信頼されているか否かを判断する（ステップＳ２２０３）。文書閲覧者から信頼されていれば（ステップＳ２２０３：Ｙｅｓ）、作成した電子文書ファイルをそのまま文書閲覧者へ配布する（ステップＳ２２０４）。文書閲覧者から信頼されていなければ（ステップＳ２２０３：Ｎｏ）、作成した電子文書ファイルをアプリ提供者に返信する（ステップＳ２２０５）。なお、電子文書ファイルをアプリ提供者に返信する際に、動作プログラムに追加したい機能があれば、その旨をアプリ提供者へ伝えてもよい。

【００７８】

次いで、アプリ提供者が実行する第２の処理について説明する。図２３は、アプリ提供者による電子文書認証のための第２の処理手順を示すフローチャートである。アプリ提供者は、返信された電子文書ファイルを取得し、電子文書ファイル内にある動作プログラムおよび動作プログラム利用情報を削除する（ステップＳ２３０１）。次に、秘密鍵・公開鍵を生成する（ステップＳ２３０２）。新たに動作プログラムおよび動作プログラム利用情報を生成し、アプリ提供者が保持する秘密鍵で動作プログラムを暗号化する（ステップＳ２３０３）。秘密鍵と対となる公開鍵は動作プログラム利用情報に保存する（ステップＳ２３０４）。暗号化した動作プログラム・動作プログラム利用情報を挿入する（ステップＳ２３０５）。この後、電子文書ファイルを再び文書作成者へ送信する。

【００７９】

次に、文書作成者が実行する第２の処理について説明する。図２４は、文書作成者による電子文書認証のための第２の処理手順を示すフローチャートである。文書作成者は、アプリ提供者から電子文書ファイルを取得し、文書内容を伝えたい文書閲覧者へ電子文書フ

ファイルを配布する（ステップＳ２４０１）。ここでは、電子文書内の動作プログラムは暗号化されているので、文書作成者は動作プログラムの改ざんを実行することができなくなる。

【００８０】

続いて、文書閲覧者が実行する処理について説明する。図２５は、文書閲覧者による電子文書認証のための処理手順を示すフローチャートである。まず、文書閲覧者は、取得した電子文書ファイルがアプリ提供者によって保証されているかを判断する（ステップＳ２５０１）。ここでは、動作プログラム利用情報に保存された復号化情報を使用して電子文書ファイル内にある動作プログラムを復号化して動作プログラムの署名情報を検証してもよいし、電子文書ファイルを起動した際に動作プログラムが暗号化されていない場合にはダイアログボックスを表示して、電子文書ファイルの安全性をアプリ提供者が保証していないことを明示してもよい。電子文書ファイルの安全性がアプリ提供者によって保証されている場合（ステップＳ２５０１：Ｙｅｓ）は、ステップＳ２５０２へ進む。電子文書ファイルの安全性がアプリ提供者によって保証されていない場合（ステップＳ２５０１：Ｎｏ）には、ステップＳ２５０３へ進む。電子文書ファイルの安全性がアプリ提供者によって保証されている場合は（ステップＳ２５０１：Ｙｅｓ）、電子文書ファイル内にある動作プログラム利用情報に保存された公開鍵情報により、動作プログラムを復号化する（ステップＳ２５０２）。次に，動作プログラムの一つである動作検証プログラムを使用して、アプリ提供者が動作プログラムに署名した署名情報を認証し、害意を持つ文書作成者による動作プログラムの改ざん検証を行う（ステップＳ２５０４）。

【００８１】

改ざんがある場合（ステップＳ２５０４：Ｙｅｓ）は、不正処理が行われたと判断し（ステップＳ２５０５）、そのまま処理を終了する（すなわち、文書閲覧を中止する）。改ざんがない場合（ステップＳ２５０４：Ｎｏ）は、コンテンツファイルを表示させるため動作プログラムの一つである文書閲覧プログラムを実行させ、コンテンツファイルを表示させる（ステップＳ２５０６）。電子文書ファイルの安全性がアプリ提供者によって保証されていない場合には（ステップＳ２５０１：Ｎｏ）、文書閲覧者に、動作プログラムの起動を判断させるダイアログボックスを表示させ、電子文書ファイルの作成者を信頼するか否かを判断させる（ステップＳ２５０３）。文書閲覧者が文書作成者を信頼する場合（ステップＳ２５０３：Ｙｅｓ）は、コンテンツファイルを表示させるため動作プログラムの一つである文書閲覧プログラムを実行させ、コンテンツファイルを表示させる（ステップＳ２５０６）。文書閲覧者が文書作成者を信頼しない場合（ステップＳ２５０３：Ｎｏ）は、動作プログラムを実行させず、コンテンツファイルが閲覧されないようにする（ステップＳ２５０７）。

【００８２】

以上のような工程を経ることにより、電子文書ファイルの安全性を文書閲覧者へ明示することによって、アプリ提供者が安全性を保証しているか否かを文書閲覧者は容易に判断できる。この結果、既知でない文書作成者から配布された有害な電子文書ファイルを誤って閲覧してしまうことを防止できる。

【００８３】

以上説明したように、本発明の電子文書ファイル、電子文書ファイル作成装置によれば、文書作成アプリケーションに電子文書の安全性を保証させることによって、電子文書の作成者を確認することなくても、有害な電子文書の閲覧が防止でき、電子文書閲覧の安全性を図ることができる。

【産業上の利用可能性】

【００８４】

以上のように、本発明にかかる電子文書ファイル、電子文書ファイル作成装置は、電子文書の安全性を電子文書作成アプリケーションソフトが保証する電子文書認証システムに有用であり、特に、コミュニケーションツール，電子商取引，会議システム，審査承認などのドキュメントワークフローシステムなどに適している。

【図面の簡単な説明】

【００８５】

【図１】実施の形態１にかかる電子文書ファイルの構造を示す図である。

【図２】動作プログラム利用情報に含まれる情報等をＸＭＬ形式で記載したフォーマットを示す図である。

【図３】実施の形態１の電子文書ファイルを用いた電子文書認証の概略を説明するための図である。

【図４】アプリ提供者による電子文書認証のための処理手順を示すフローチャートである。

【図５】文書作成者による電子文書認証のための処理手順を示すフローチャートである。

【図６】文書閲覧者による電子文書認証のための処理手順を示すフローチャートである。

【図７】動作プログラムの実行を制限させる情報の一例を示す図表である。

【図８】実施の形態１にかかる電子文書ファイルを用いて行う電子文書認証システムの全体構成図である。

【図９】実施の形態１にかかる電子文書ファイルを用いて行う電子文書認証システムの全体構成図である。

【図１０】電子文書認証システム内でアプリ提供者・文書作成者・文書閲覧者が使用する代表的なハードウェア構成図である。

【図１１】アプリ提供者によるコンテンツファイルの改ざん検証のための処理手順を示すフローチャートである。

【図１２】文書作成者によるコンテンツファイルの改ざん検証のための処理手順を示すフローチャートである。

【図１３】文書閲覧者によるコンテンツファイルの改ざん検証のための処理手順を示すフローチャートである。

【図１４】アプリ提供者によるコンテンツ著作権保護のための処理手順を示すフローチャートである。

【図１５】文書作成者によるコンテンツ著作権保護のための処理手順を示すフローチャートである。

【図１６】文書閲覧者によるコンテンツ著作権保護のための処理手順を示すフローチャートである。

【図１７】著作権の使用許可願い通知を行う場合の処理手順を示すフローチャートである。

【図１８】使用許可願い通知を行う場合に表示される、コンテンツファイル利用情報のフォーマットを示す図である。

【図１９】実施の形態３にかかる電子文書ファイルを用いて行う電子文書認証システムの全体構成図である。

【図２０】実施の形態３にかかる電子文書ファイルを用いて行う電子文書認証の概略を説明するための図である。

【図２１】アプリ提供者による電子文書認証のための第１の処理手順を示すフローチャートである。

【図２２】文書作成者による電子文書認証のための第１の処理手順を示すフローチャートである。

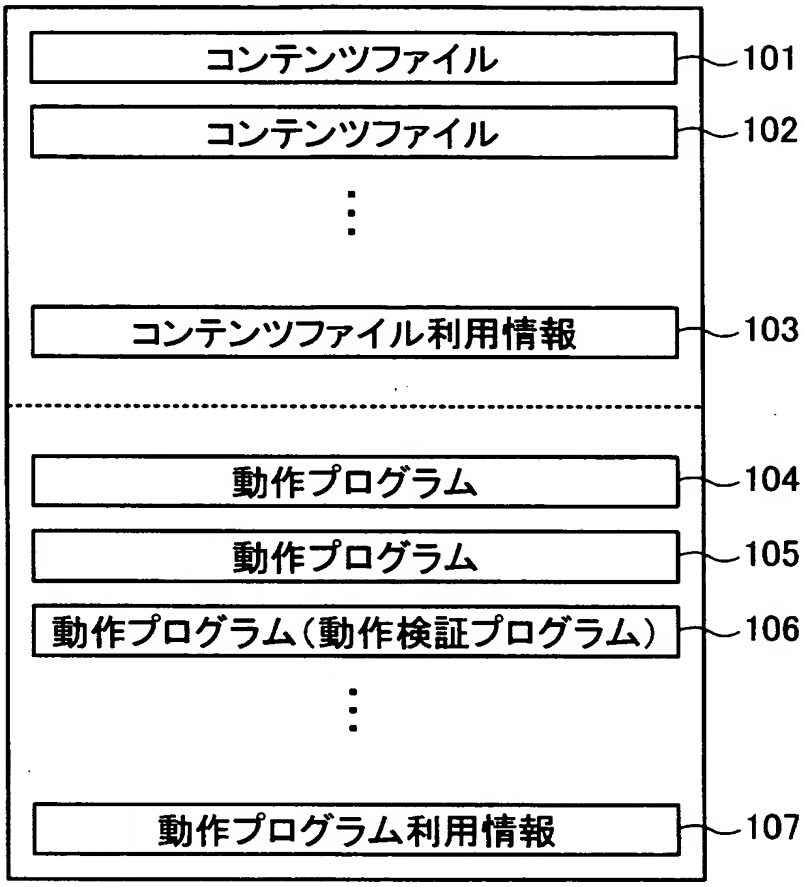
【図２３】アプリ提供者による電子文書認証のための第２の処理手順を示すフローチャートである。

【図２４】文書作成者による電子文書認証のための第２の処理手順を示すフローチャートである。

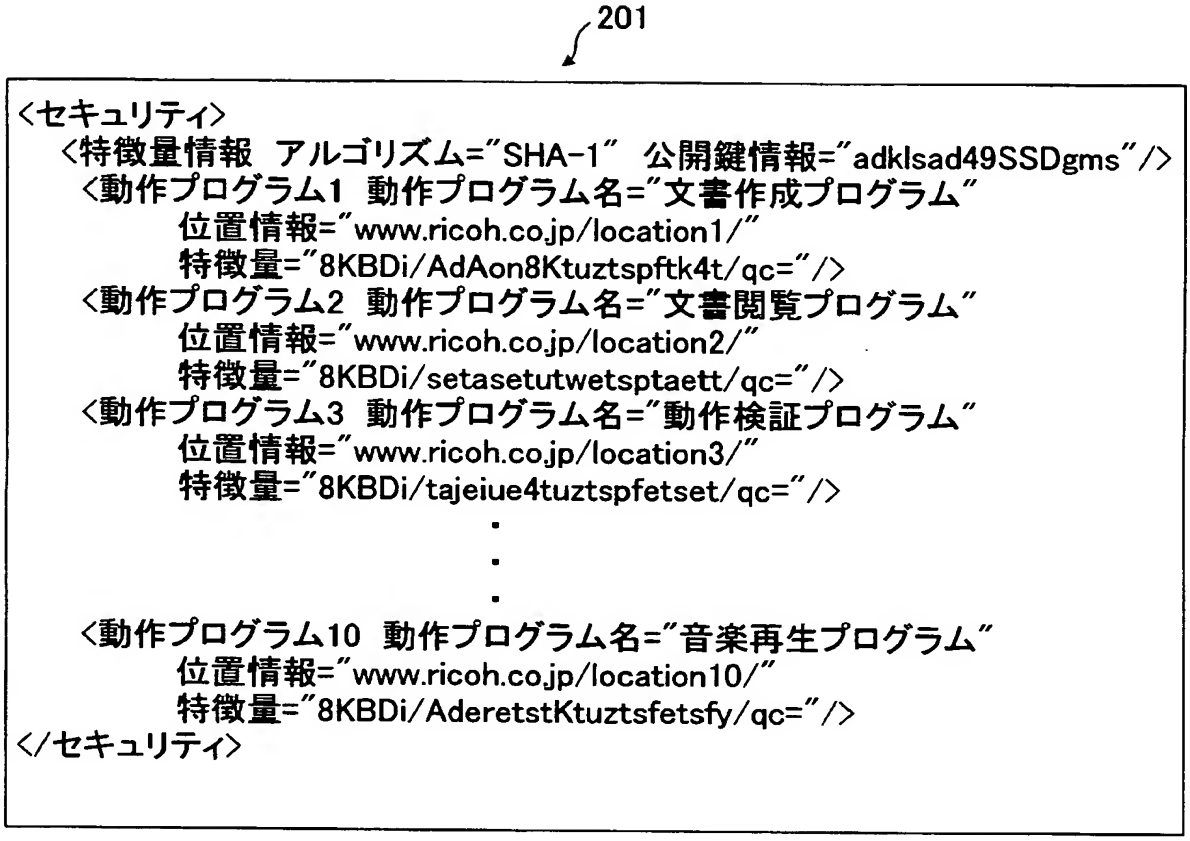
【図２５】文書閲覧者による電子文書認証のための処理手順を示すフローチャートである。

【符号の説明】
【００８６】
１０１，１０２　コンテンツファイル
１０３　コンテンツファイル利用情報
１０４，１０５，１０６　動作プログラム
１０７　動作プログラム利用情報
２０１，１８０１，１８０２　フォーマット
８０１，９０１，１９０１　アプリ提供者
８０２，９０２，１９０２　文書作成者
８０３，９０３，１９０３　文書閲覧者
１００１　ＣＰＵ
１００２　ＲＡＭ
１００３　ＨＤＤ
１００４　入力Ｉ／Ｆ
１００５　表示Ｉ／Ｆ
１００６　ディスプレイ
１００７　記録装置
１００８　外部Ｉ／Ｆ
１００９　バス

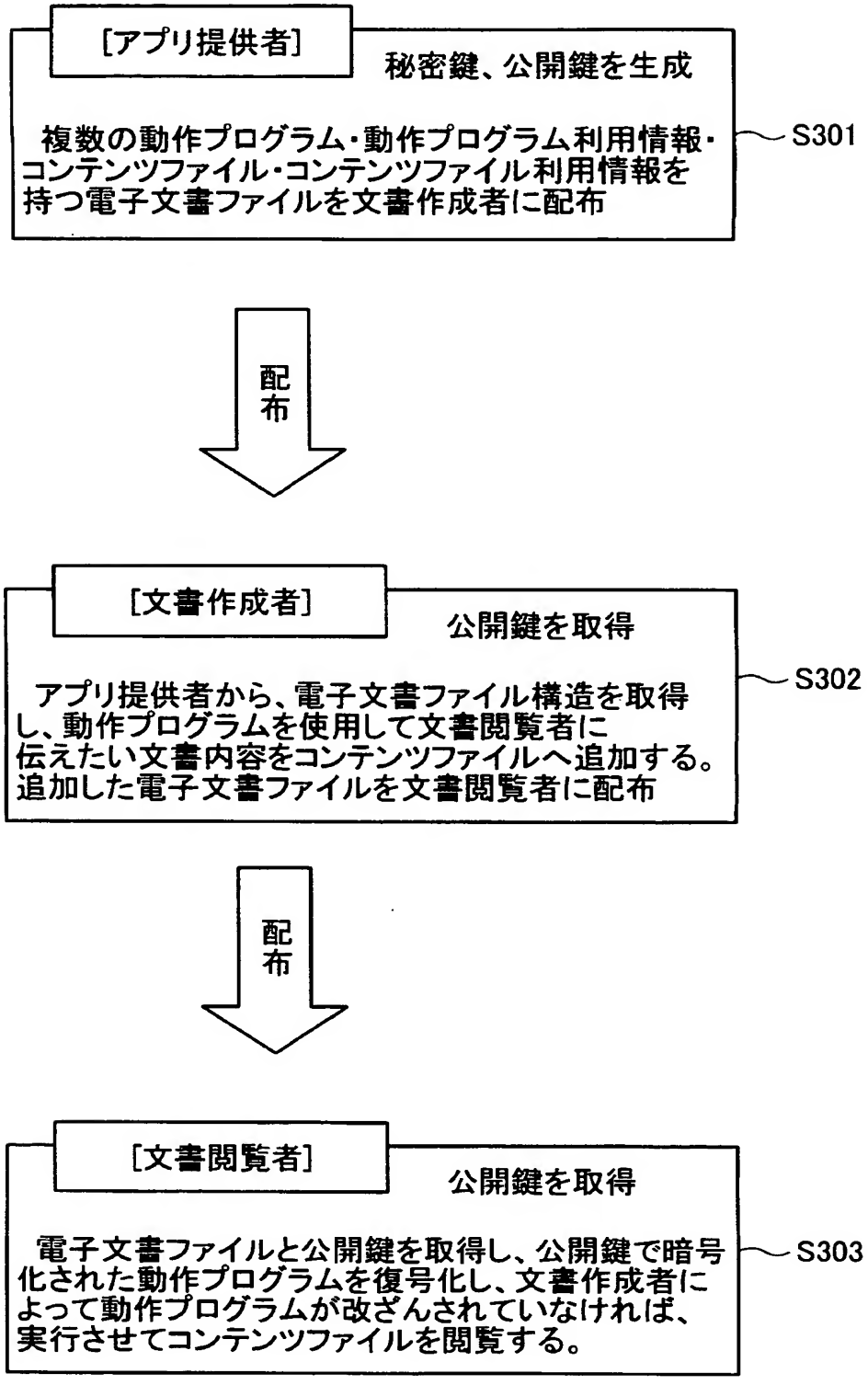
【書類名】図面
【図１】


コンテンツファイル
101
コンテンツファイル
102
コンテンツファイル利用情報
103
動作プログラム
104
動作プログラム
105
動作プログラム（動作検証プログラム）
106
動作プログラム利用情報
107

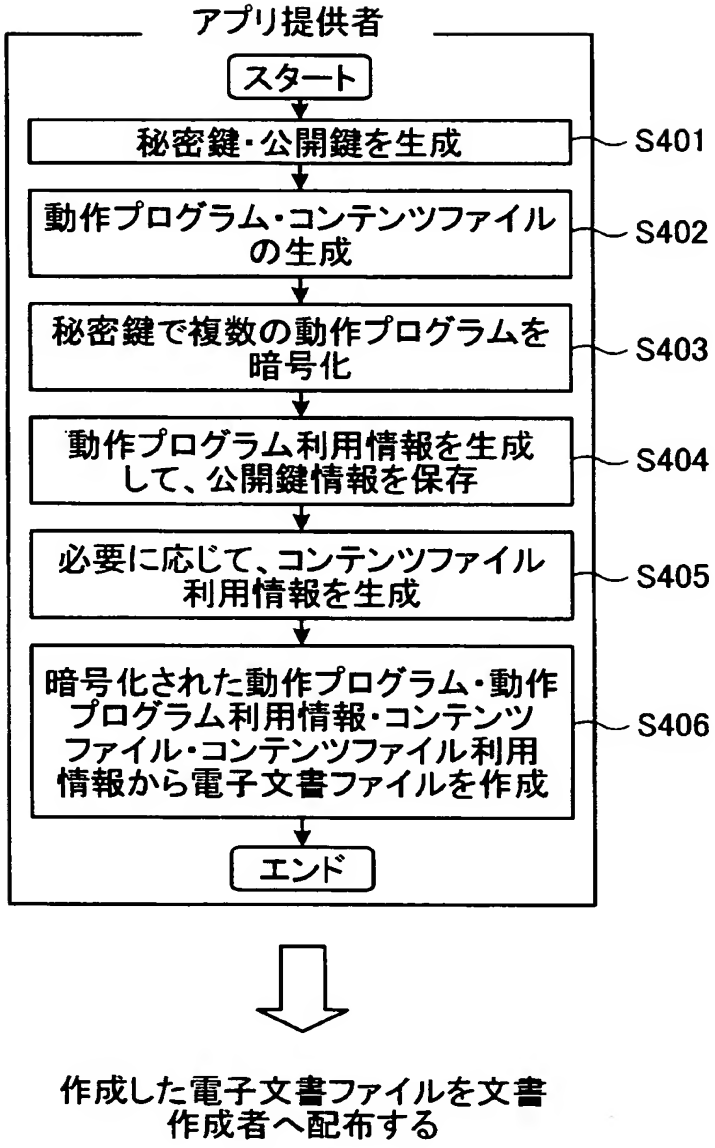
【図２】


201
<セキュリティ>
<特徴量情報 アルゴリズム="SHA-1" 公開鍵情報="adklsad49SSDgms"/>
<動作プログラム1 動作プログラム名="文書作成プログラム"
位置情報="www.ricoh.co.jp/location1/"
特徴量="8KBDi/AdAon8Ktuztspftk4t/qc="/>
<動作プログラム2 動作プログラム名="文書閲覧プログラム"
位置情報="www.ricoh.co.jp/location2/"
特徴量="8KBDi/setasetutwetsptaett/qc="/>
<動作プログラム3 動作プログラム名="動作検証プログラム"
位置情報="www.ricoh.co.jp/location3/"
特徴量="8KBDi/tajeiue4tuztspfetset/qc="/>
<動作プログラム10 動作プログラム名="音楽再生プログラム"
位置情報="www.ricoh.co.jp/location10/"
特徴量="8KBDi/AderetstKtuztsfetsfy/qc="/>
</セキュリティ>

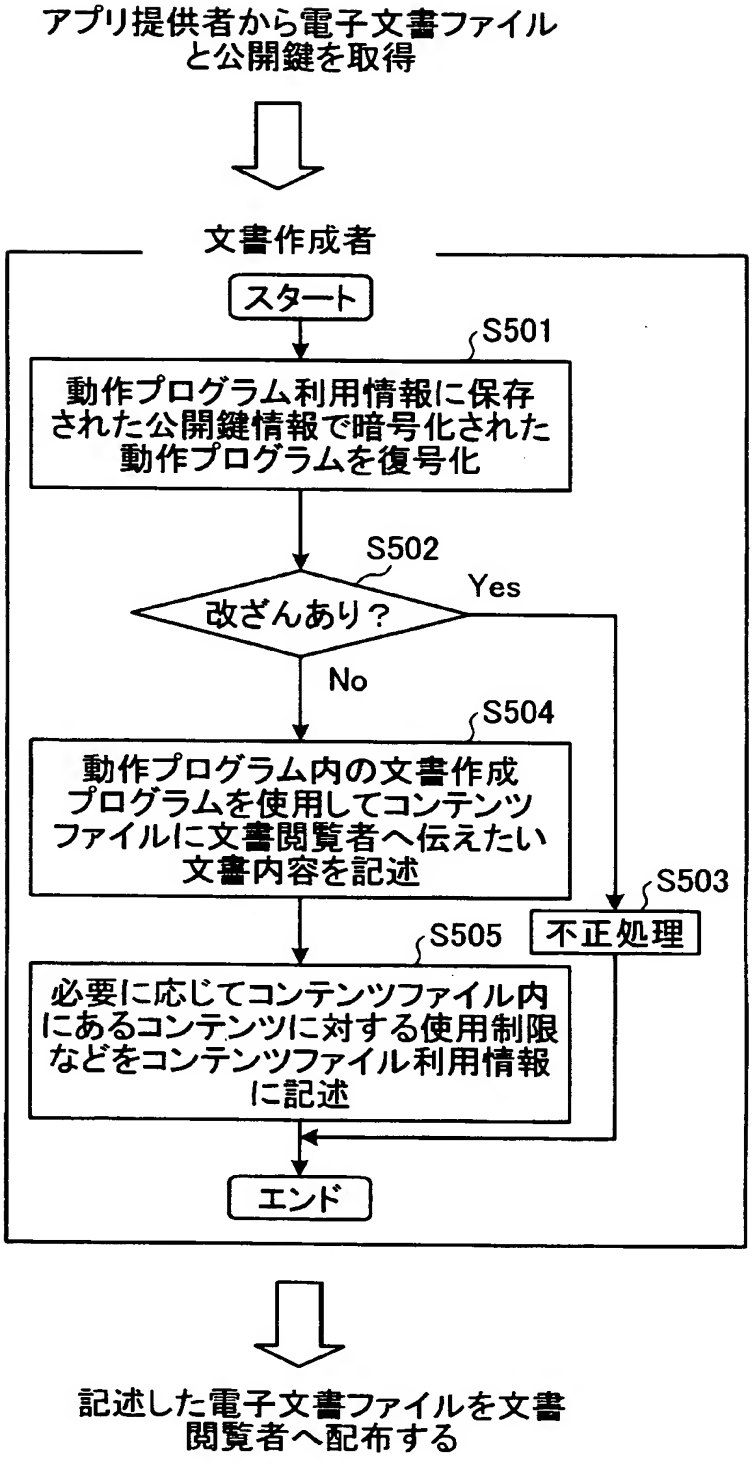
【図３】


[アプリ提供者]
秘密鍵、公開鍵を生成
複数の動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報を持つ電子文書ファイルを文書作成者に配布
S301
配布
[文書作成者]
公開鍵を取得
アプリ提供者から、電子文書ファイル構造を取得し、動作プログラムを使用して文書閲覧者に伝えたい文書内容をコンテンツファイルへ追加する。追加した電子文書ファイルを文書閲覧者に配布
S302
配布
[文書閲覧者]
公開鍵を取得
電子文書ファイルと公開鍵を取得し、公開鍵で暗号化された動作プログラムを復号化し、文書作成者によって動作プログラムが改ざんされていなければ、実行させてコンテンツファイルを閲覧する。
S303

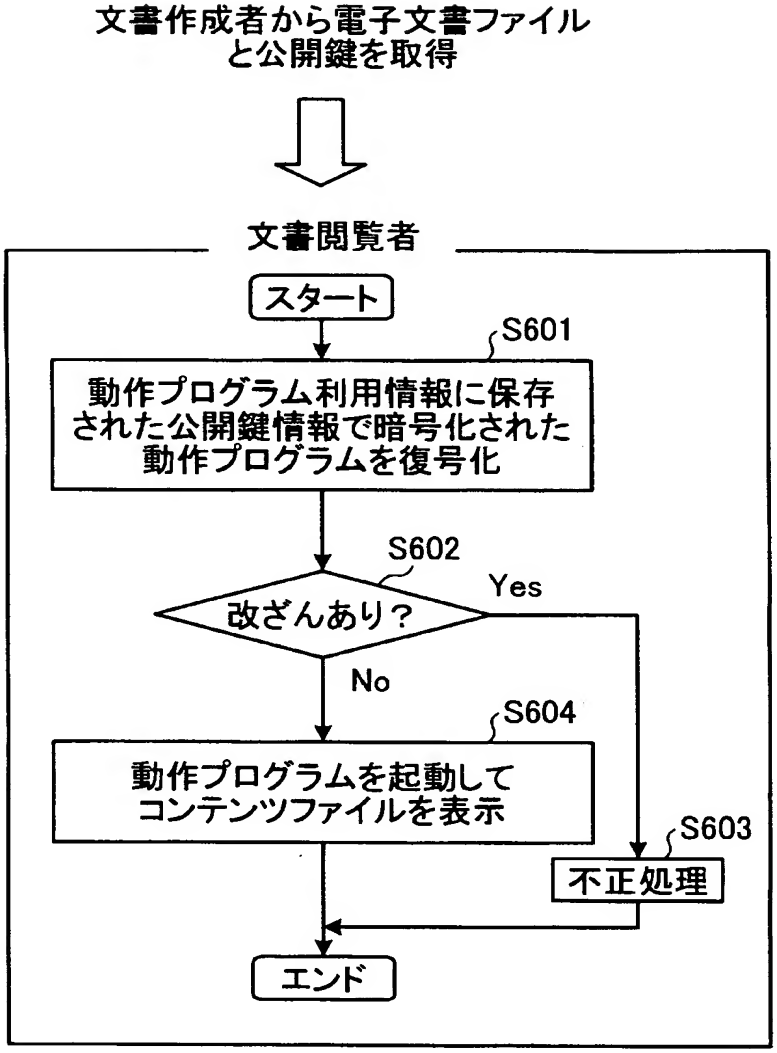
【図４】


アプリ提供者
スタート
秘密鍵・公開鍵を生成
S401
動作プログラム・コンテンツファイルの生成
S402
秘密鍵で複数の動作プログラムを暗号化
S403
動作プログラム利用情報を生成して、公開鍵情報を保存
S404
必要に応じて、コンテンツファイル利用情報を生成
S405
暗号化された動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から電子文書ファイルを作成
S406
エンド
作成した電子文書ファイルを文書作成者へ配布する

【図５】


アプリ提供者から電子文書ファイル
と公開鍵を取得
文書作成者
スタート
S501
動作プログラム利用情報に保存
された公開鍵情報で暗号化された
動作プログラムを復号化
S502
改ざんあり？
Yes
No
S504
動作プログラム内の文書作成
プログラムを使用してコンテンツ
ファイルに文書閲覧者へ伝えたい
文書内容を記述
S503
不正処理
S505
必要に応じてコンテンツファイル内
にあるコンテンツに対する使用制限
などをコンテンツファイル利用情報
に記述
エンド
記述した電子文書ファイルを文書
閲覧者へ配布する

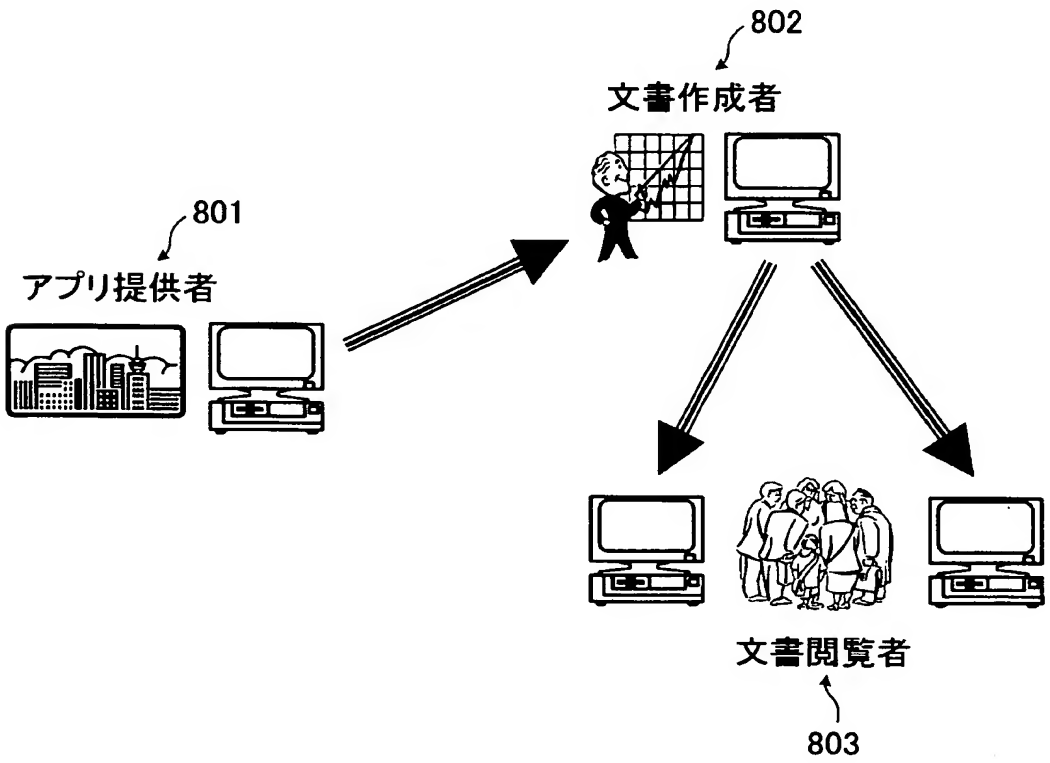
【図６】


文書作成者から電子文書ファイル
と公開鍵を取得
文書閲覧者
スタート
S601
動作プログラム利用情報に保存
された公開鍵情報で暗号化された
動作プログラムを復号化
S602
改ざんあり？
Yes
No
S604
動作プログラムを起動して
コンテンツファイルを表示
S603
不正処理
エンド

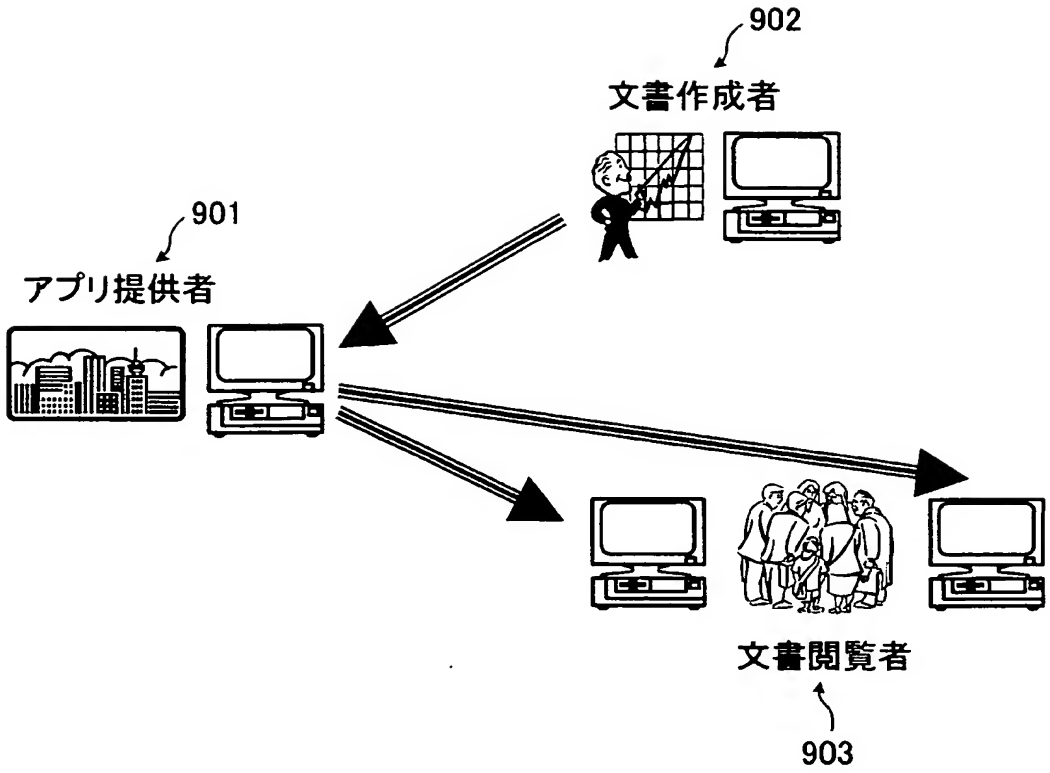
【図７】

動作権限の種類	動作権限モード				
カプセル化文書内のファイルの読込み	×	○	○	○	
コンピュータ内のファイルの読込み	×	×	○	○	
カプセル化文書内のファイルの読み書き	×	×	×	○	
コンピュータ内のファイルの読み書き	×	×	×	×	・・・・・・・・・・・・
ネットワークのパケットの受信	×	×	×	×	
ネットワークのパケットの送信	×	×	×	×	
⋮					

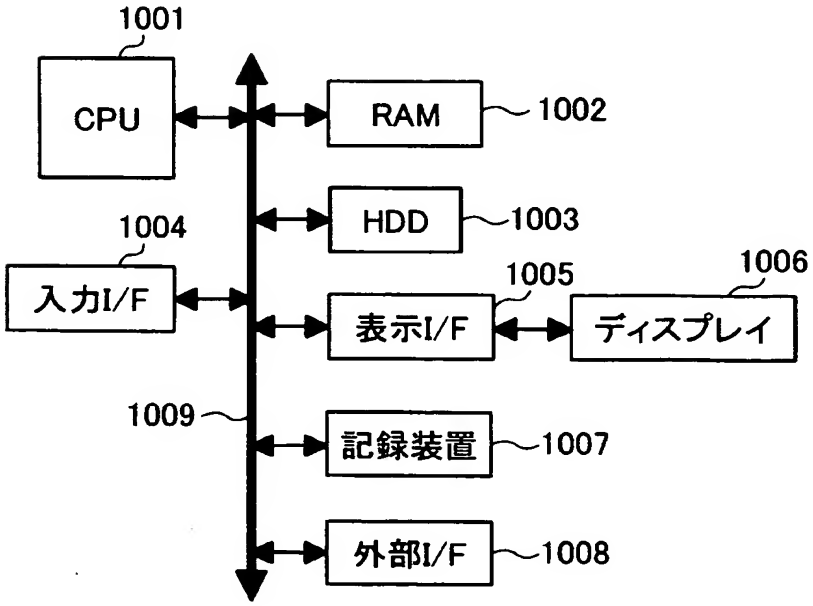
【図８】



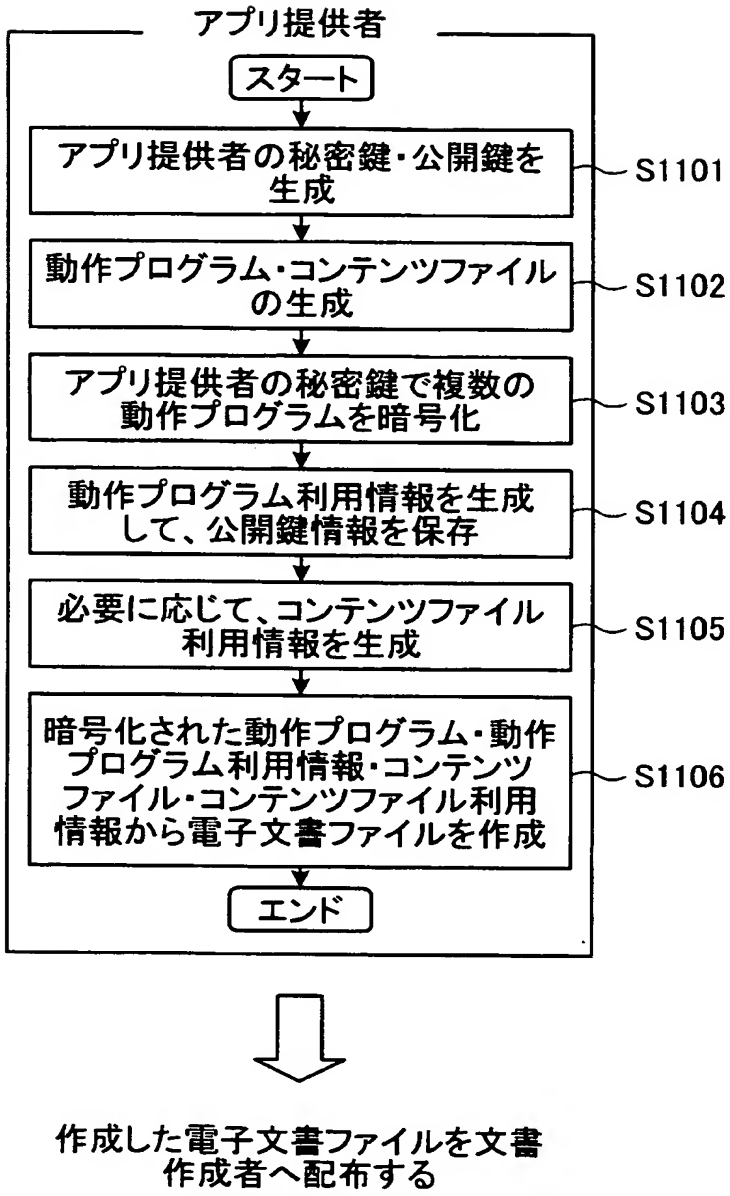
【図９】


902
文書作成者
901
アプリ提供者
文書閲覧者
903

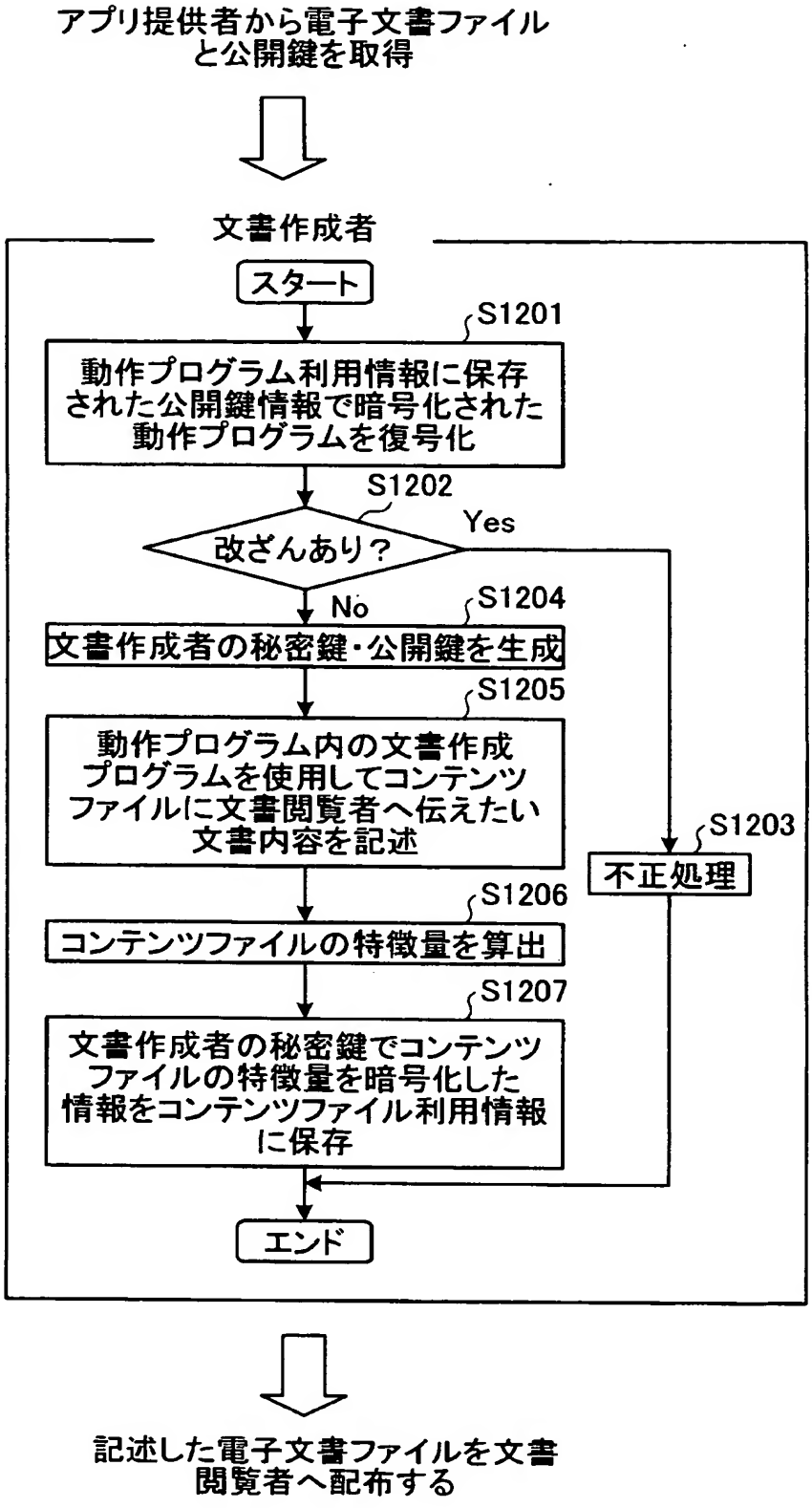
【図１０】


1001
CPU
RAM
1002
HDD
1003
1004
入力I/F
1005
1006
表示I/F
ディスプレイ
1009
記録装置
1007
外部I/F
1008

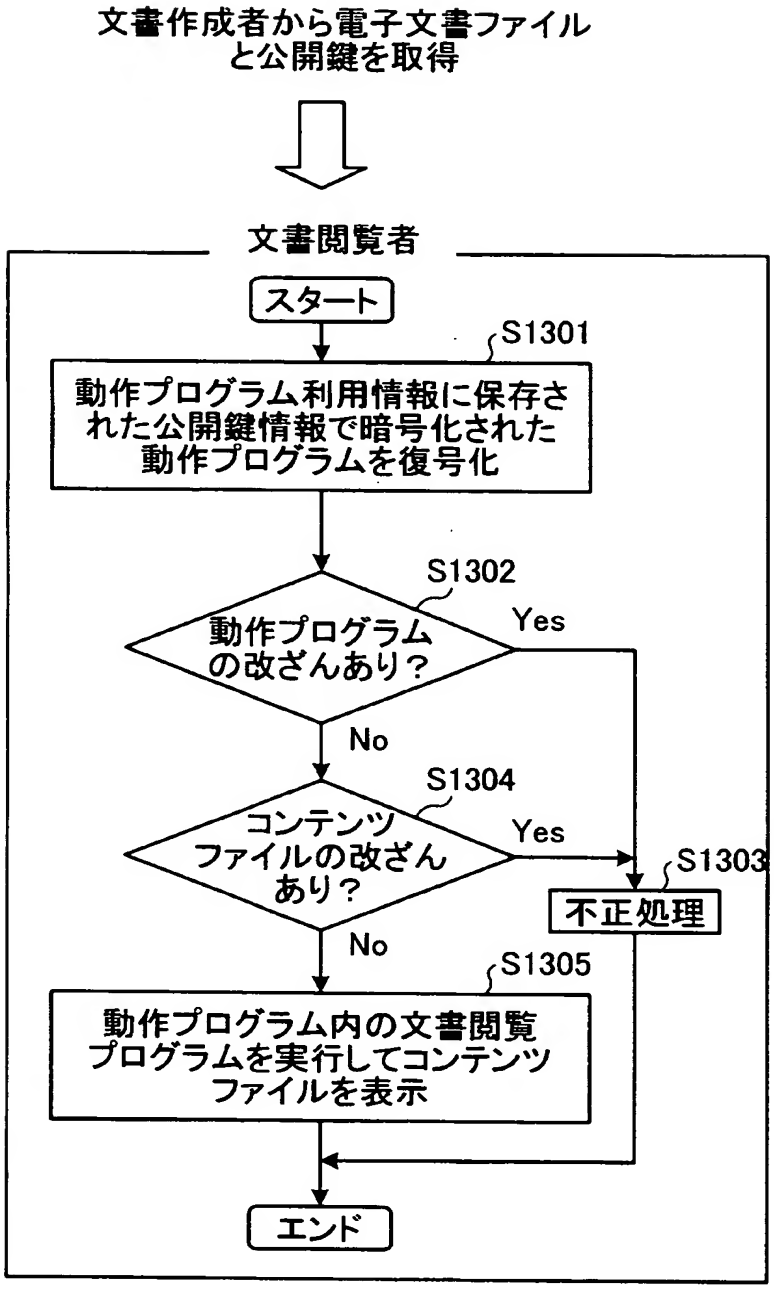
【図１１】


アプリ提供者
スタート
アプリ提供者の秘密鍵・公開鍵を生成
S1101
動作プログラム・コンテンツファイルの生成
S1102
アプリ提供者の秘密鍵で複数の動作プログラムを暗号化
S1103
動作プログラム利用情報を生成して、公開鍵情報を保存
S1104
必要に応じて、コンテンツファイル利用情報を生成
S1105
暗号化された動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から電子文書ファイルを作成
S1106
エンド
作成した電子文書ファイルを文書作成者へ配布する

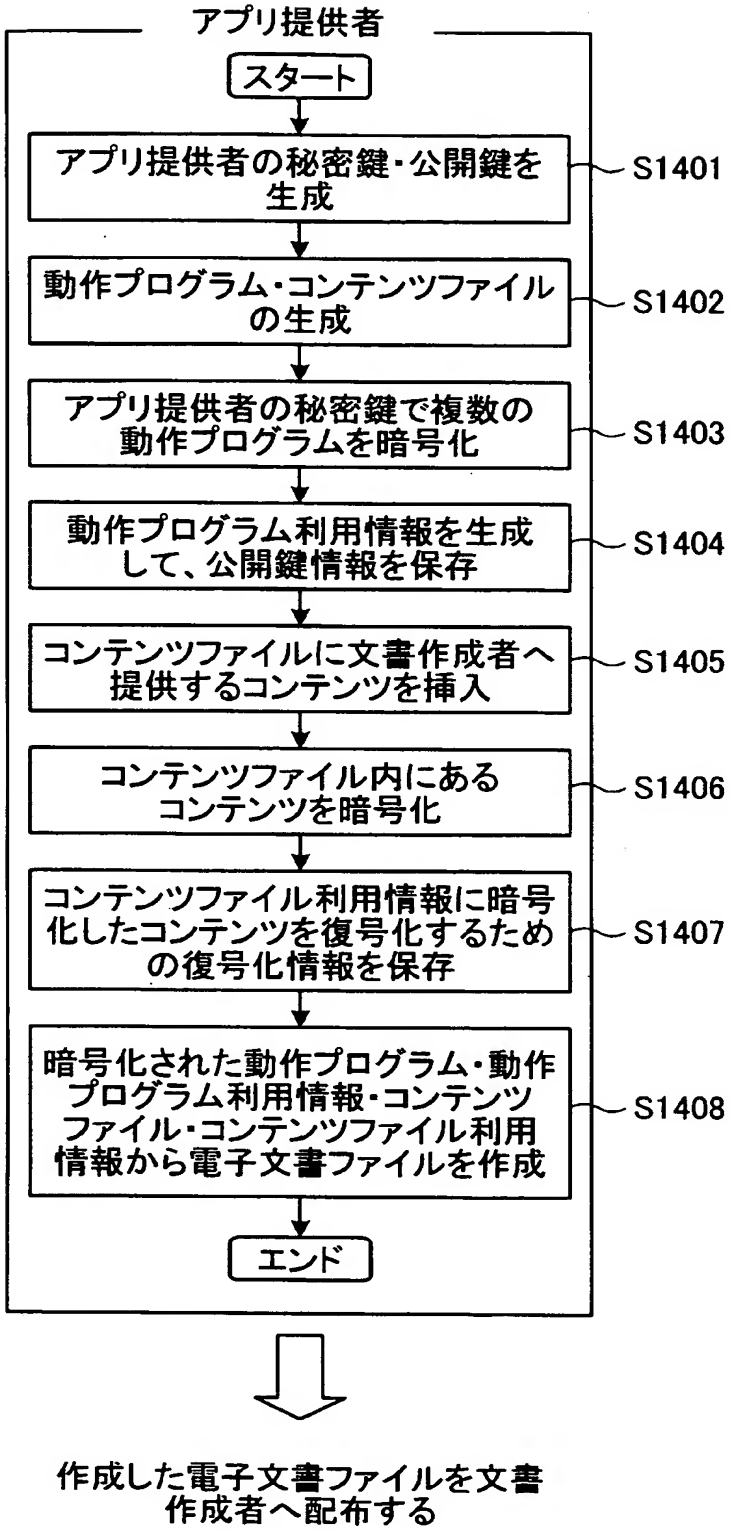
【図１２】


アプリ提供者から電子文書ファイル
と公開鍵を取得
文書作成者
スタート
S1201
動作プログラム利用情報に保存
された公開鍵情報で暗号化された
動作プログラムを復号化
S1202
改ざんあり？
Yes
No
S1204
文書作成者の秘密鍵・公開鍵を生成
S1205
動作プログラム内の文書作成
プログラムを使用してコンテンツ
ファイルに文書閲覧者へ伝えたい
文書内容を記述
S1203
不正処理
S1206
コンテンツファイルの特徴量を算出
S1207
文書作成者の秘密鍵でコンテンツ
ファイルの特徴量を暗号化した
情報をコンテンツファイル利用情報
に保存
エンド
記述した電子文書ファイルを文書
閲覧者へ配布する

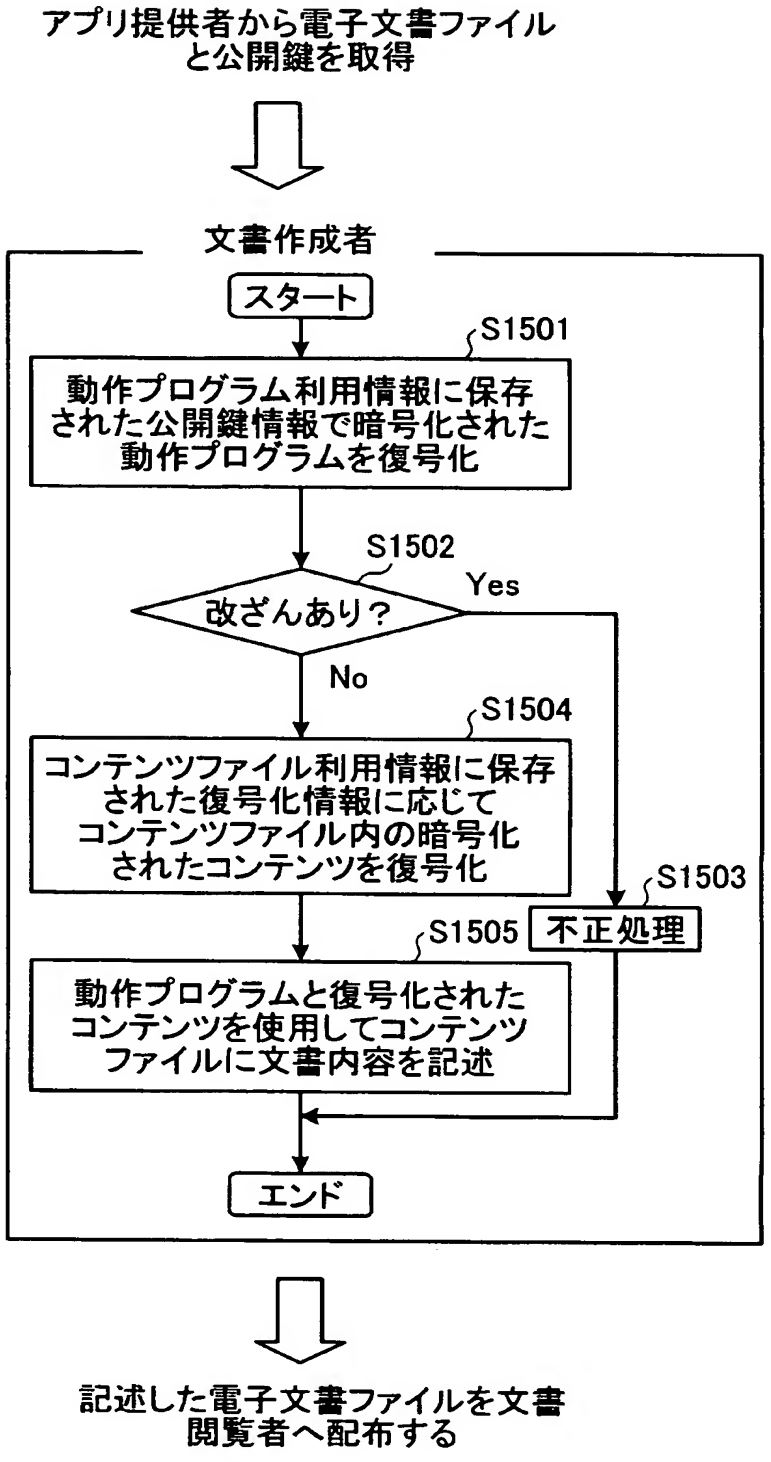
【図１３】


文書作成者から電子文書ファイル
と公開鍵を取得
文書閲覧者
スタート
S1301
動作プログラム利用情報に保存さ
れた公開鍵情報で暗号化された
動作プログラムを復号化
S1302
動作プログラム
の改ざんあり？
Yes
No
S1304
コンテンツ
ファイルの改ざん
あり？
Yes
S1303
不正処理
No
S1305
動作プログラム内の文書閲覧
プログラムを実行してコンテンツ
ファイルを表示
エンド

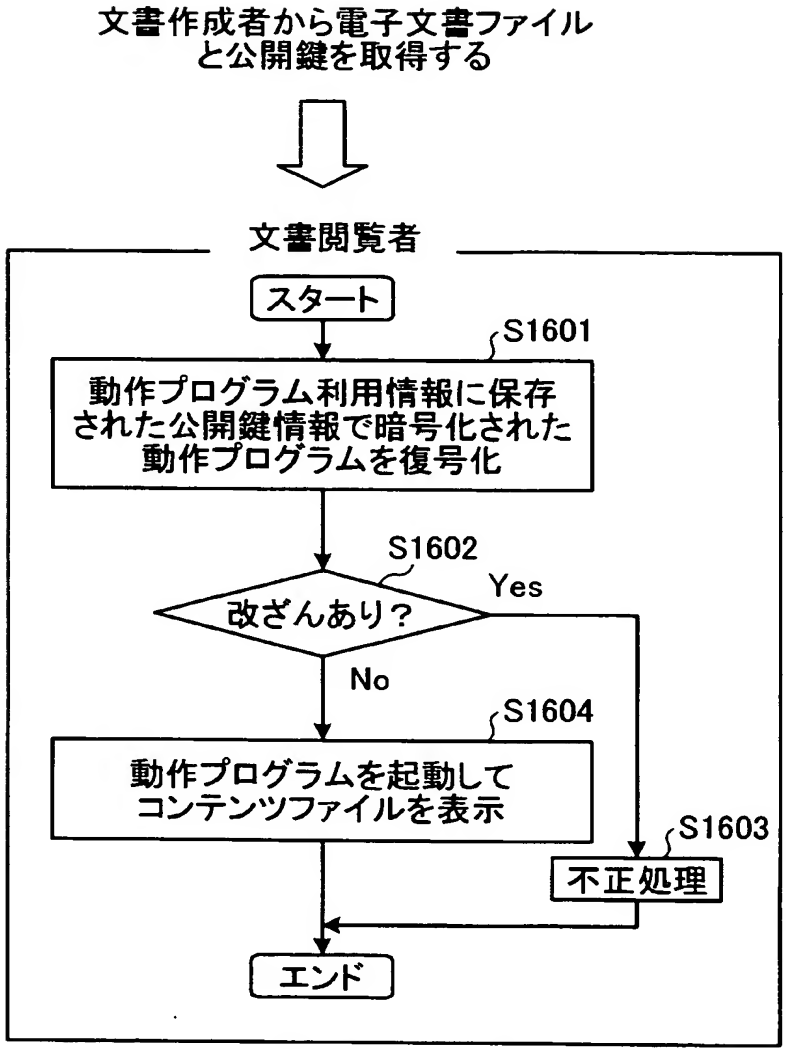
【図１４】


アプリ提供者
スタート
アプリ提供者の秘密鍵・公開鍵を生成
S1401
動作プログラム・コンテンツファイルの生成
S1402
アプリ提供者の秘密鍵で複数の動作プログラムを暗号化
S1403
動作プログラム利用情報を生成して、公開鍵情報を保存
S1404
コンテンツファイルに文書作成者へ提供するコンテンツを挿入
S1405
コンテンツファイル内にあるコンテンツを暗号化
S1406
コンテンツファイル利用情報に暗号化したコンテンツを復号化するための復号化情報を保存
S1407
暗号化された動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から電子文書ファイルを作成
S1408
エンド
作成した電子文書ファイルを文書作成者へ配布する

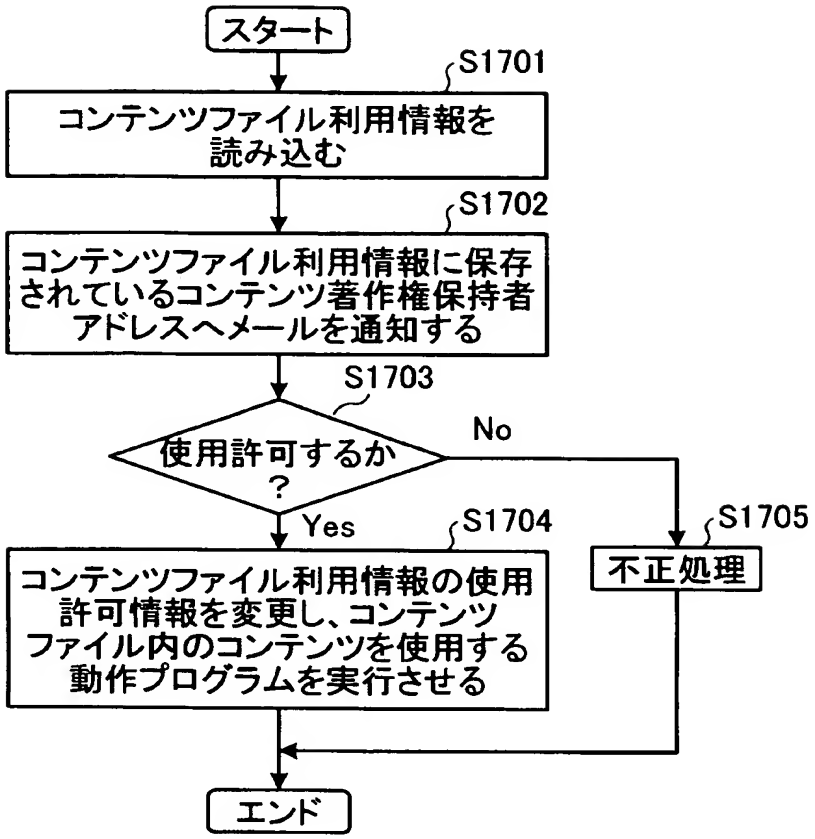
【図１５】


アプリ提供者から電子文書ファイル
と公開鍵を取得
文書作成者
スタート
S1501
動作プログラム利用情報に保存
された公開鍵情報で暗号化された
動作プログラムを復号化
S1502
改ざんあり？
Yes
No
S1504
コンテンツファイル利用情報に保存
された復号化情報に応じて
コンテンツファイル内の暗号化
されたコンテンツを復号化
S1503
不正処理
S1505
動作プログラムと復号化された
コンテンツを使用してコンテンツ
ファイルに文書内容を記述
エンド
記述した電子文書ファイルを文書
閲覧者へ配布する

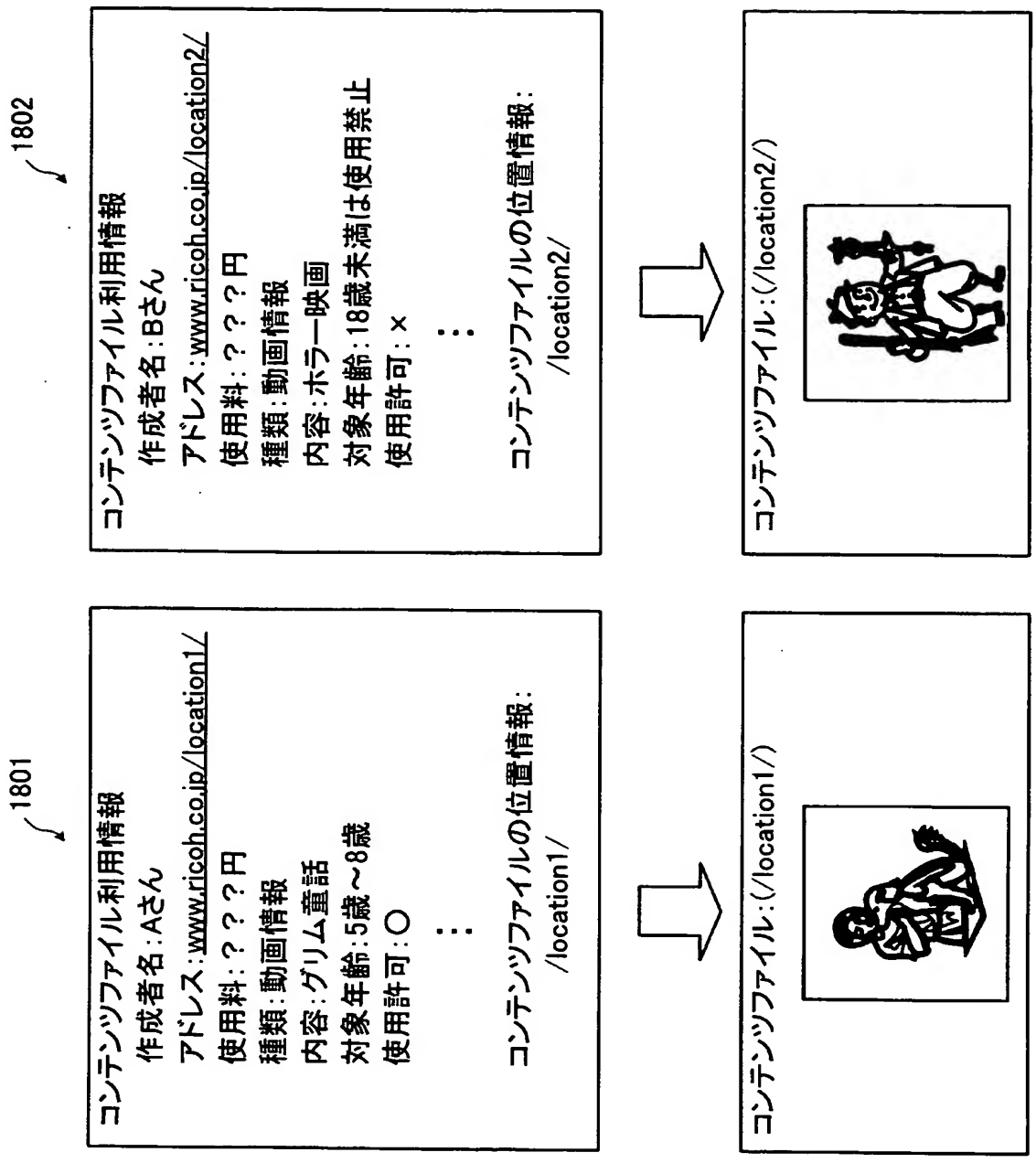
【図１６】


文書作成者から電子文書ファイル
と公開鍵を取得する
文書閲覧者
スタート
S1601
動作プログラム利用情報に保存
された公開鍵情報で暗号化された
動作プログラムを復号化
S1602
改ざんあり？
Yes
No
S1604
動作プログラムを起動して
コンテンツファイルを表示
S1603
不正処理
エンド

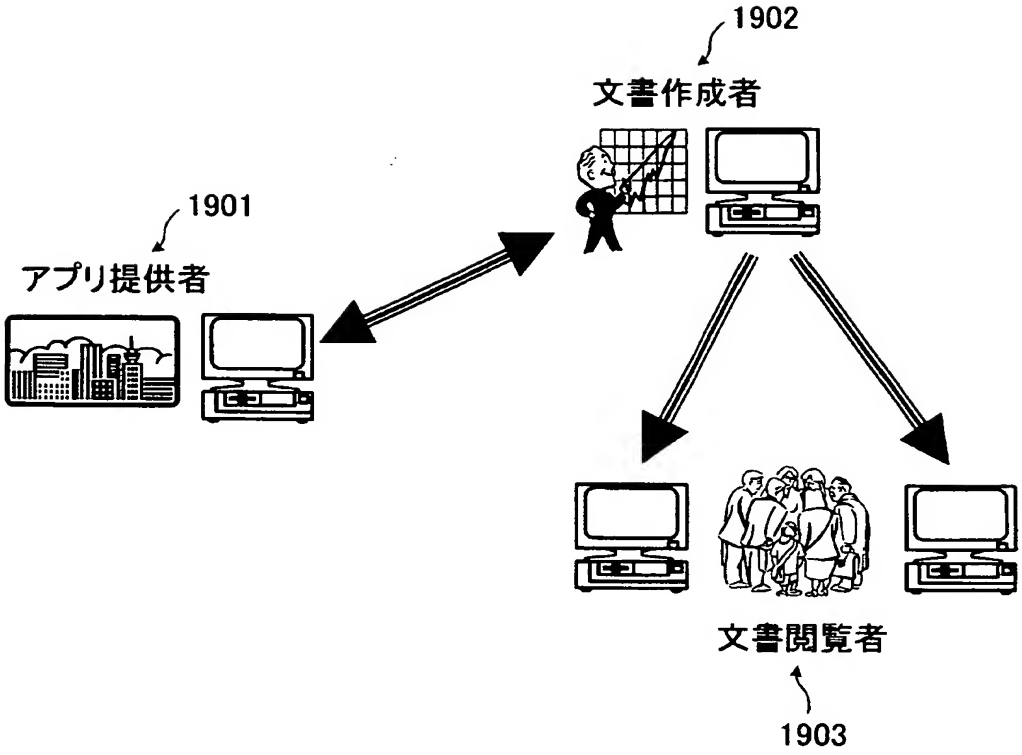
【図１７】


スタート
S1701
コンテンツファイル利用情報を読み込む
S1702
コンテンツファイル利用情報に保存されているコンテンツ著作権保持者アドレスへメールを通知する
S1703
使用許可するか？
No
Yes
S1704
コンテンツファイル利用情報の使用許可情報を変更し、コンテンツファイル内のコンテンツを使用する動作プログラムを実行させる
S1705
不正処理
エンド

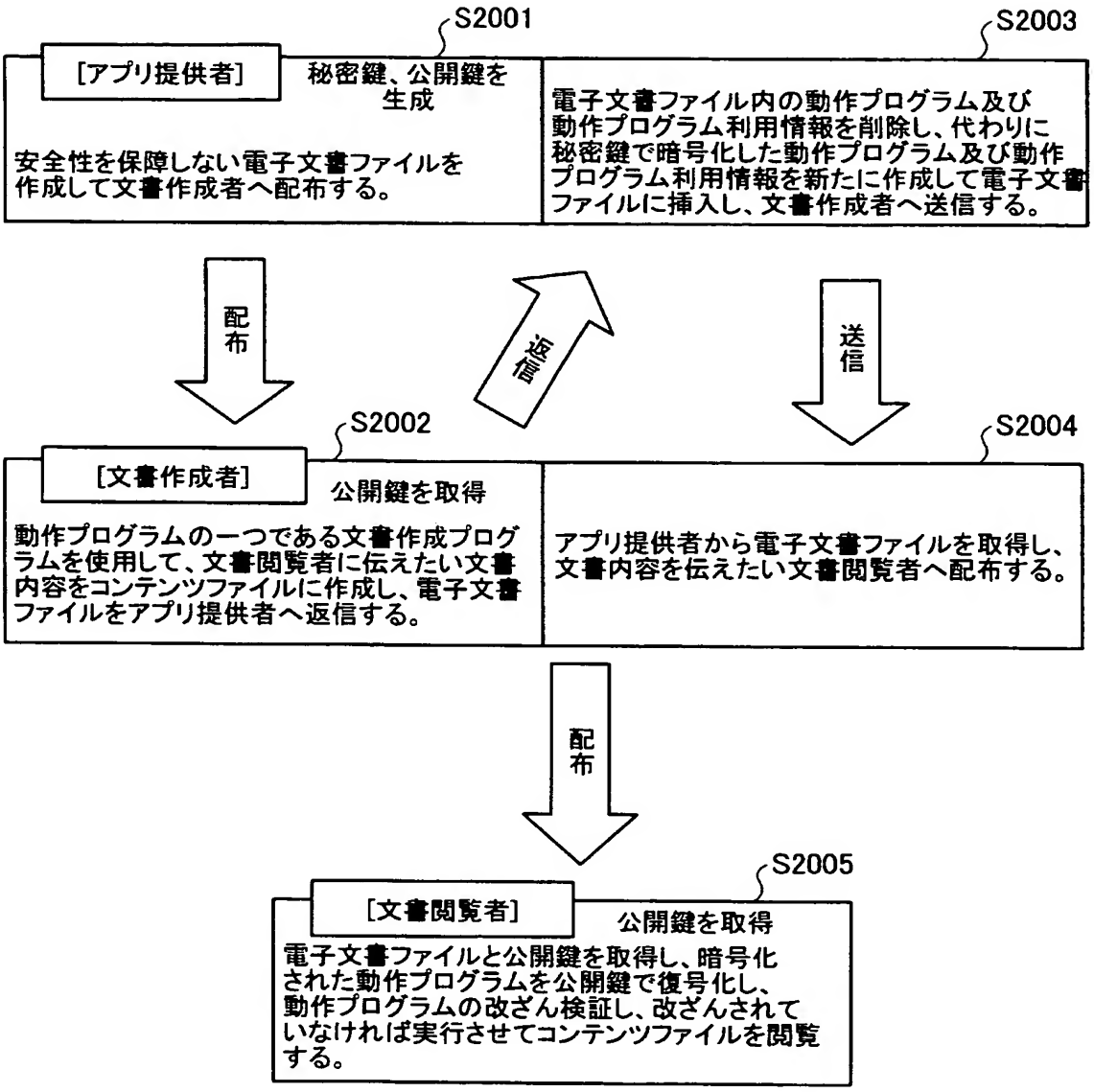
【図１８】


1801
コンテンツファイル利用情報
作成者名:Aさん
アドレス:www.ricoh.co.jp/location1/
使用料:？？？円
種類:動画情報
内容:グリム童話
対象年齢:5歳～8歳
使用許可:○
：
コンテンツファイルの位置情報:
/location1/
コンテンツファイル:(/location1/)
1802
コンテンツファイル利用情報
作成者名:Bさん
アドレス:www.ricoh.co.jp/location2/
使用料:？？？円
種類:動画情報
内容:ホラー映画
対象年齢:18歳未満は使用禁止
使用許可:×
：
コンテンツファイルの位置情報:
/location2/
コンテンツファイル:(/location2/)

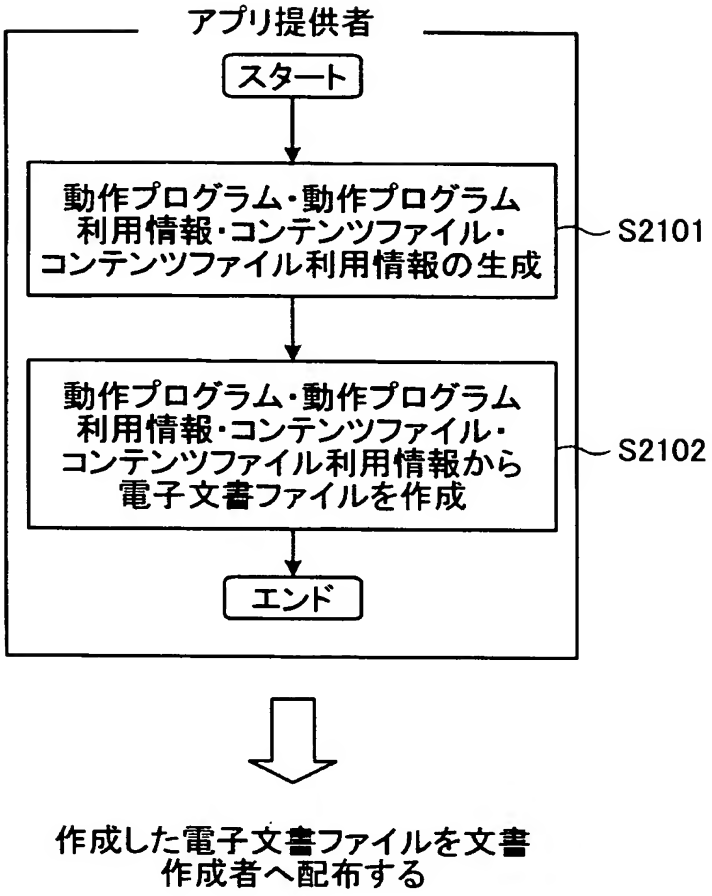
【図１９】



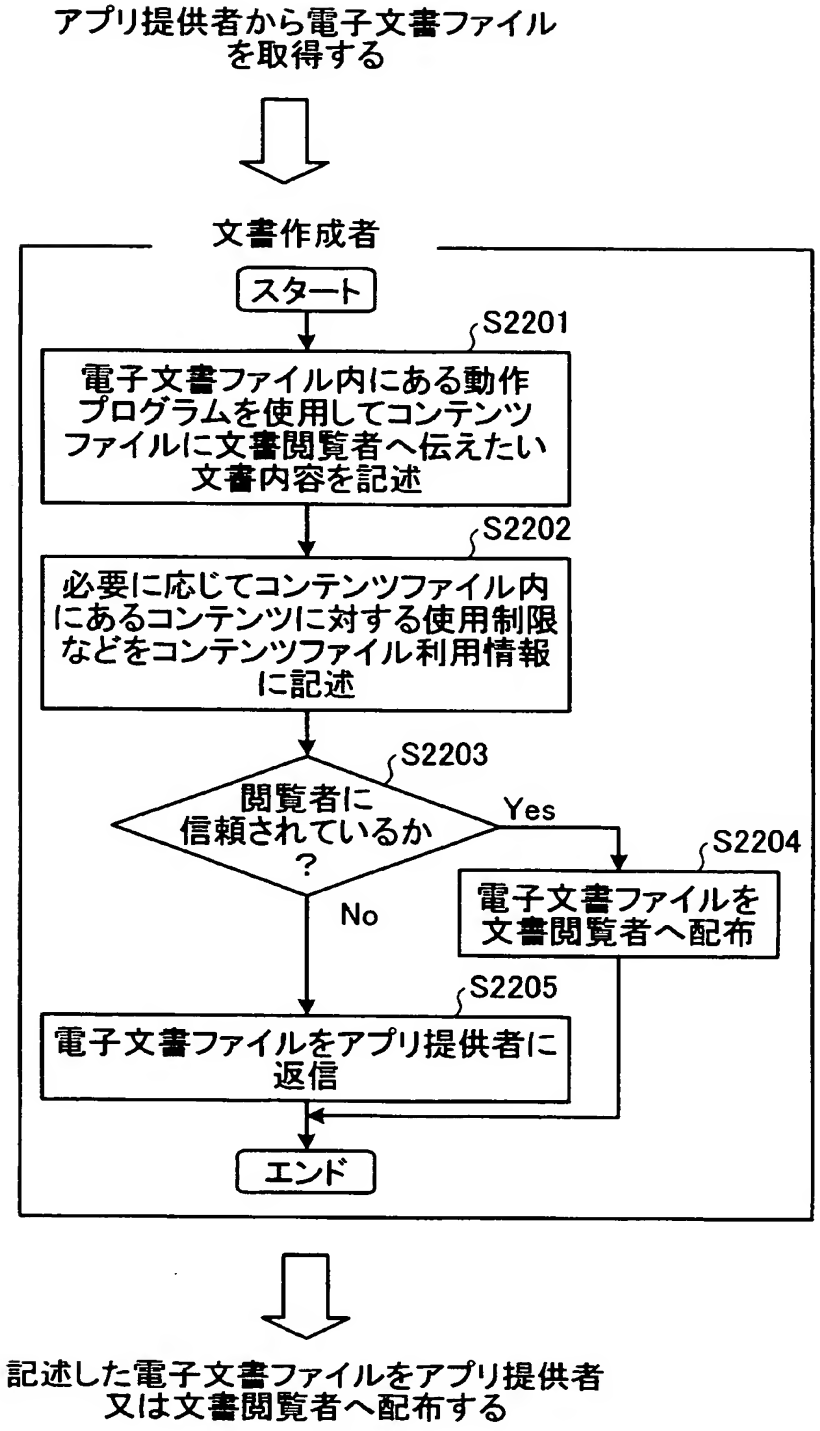
【図２０】


S2001
[アプリ提供者]
秘密鍵、公開鍵を生成
安全性を保障しない電子文書ファイルを作成して文書作成者へ配布する。
S2003
電子文書ファイル内の動作プログラム及び動作プログラム利用情報を削除し、代わりに秘密鍵で暗号化した動作プログラム及び動作プログラム利用情報を新たに作成して電子文書ファイルに挿入し、文書作成者へ送信する。
配布
返信
送信
S2002
[文書作成者]
公開鍵を取得
動作プログラムの一つである文書作成プログラムを使用して、文書閲覧者に伝えたい文書内容をコンテンツファイルに作成し、電子文書ファイルをアプリ提供者へ返信する。
S2004
アプリ提供者から電子文書ファイルを取得し、文書内容を伝えたい文書閲覧者へ配布する。
配布
S2005
[文書閲覧者]
公開鍵を取得
電子文書ファイルと公開鍵を取得し、暗号化された動作プログラムを公開鍵で復号化し、動作プログラムの改ざん検証し、改ざんされていなければ実行させてコンテンツファイルを閲覧する。

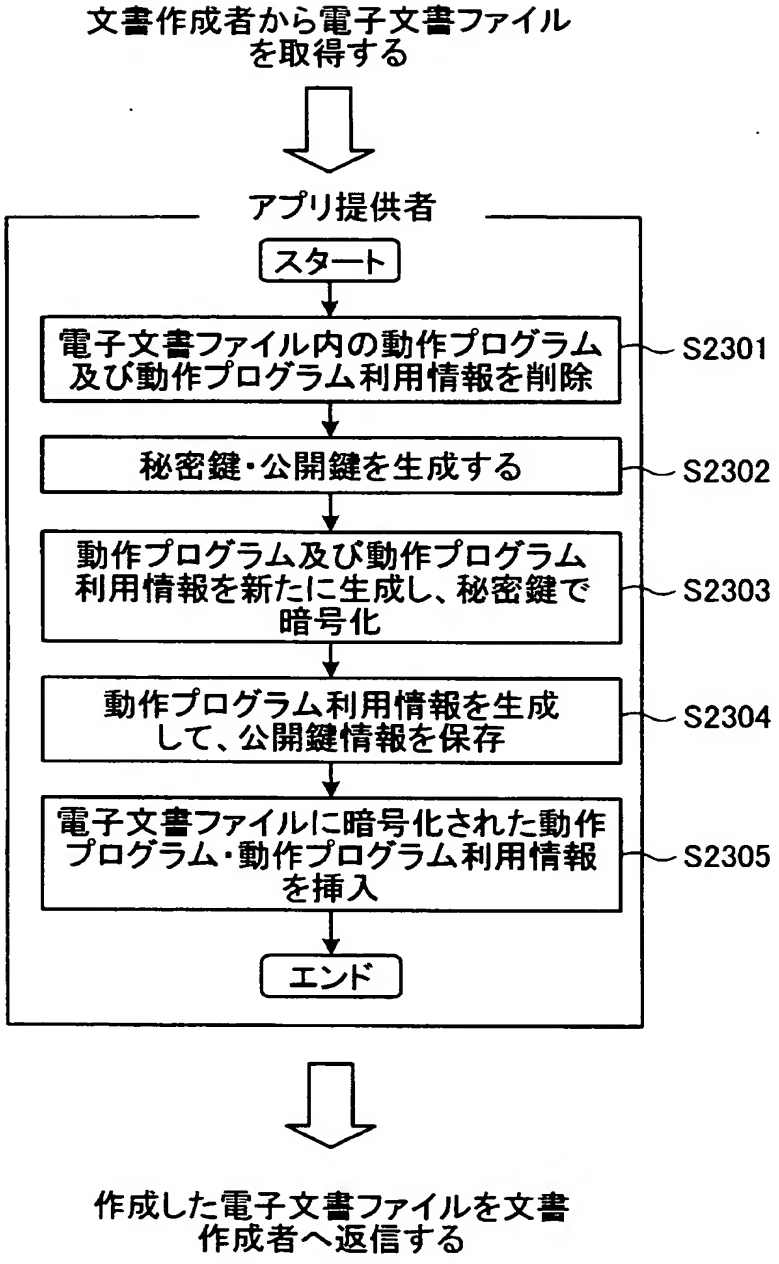
【図２１】


アプリ提供者
スタート
動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報の生成
S2101
動作プログラム・動作プログラム利用情報・コンテンツファイル・コンテンツファイル利用情報から電子文書ファイルを作成
S2102
エンド
作成した電子文書ファイルを文書作成者へ配布する

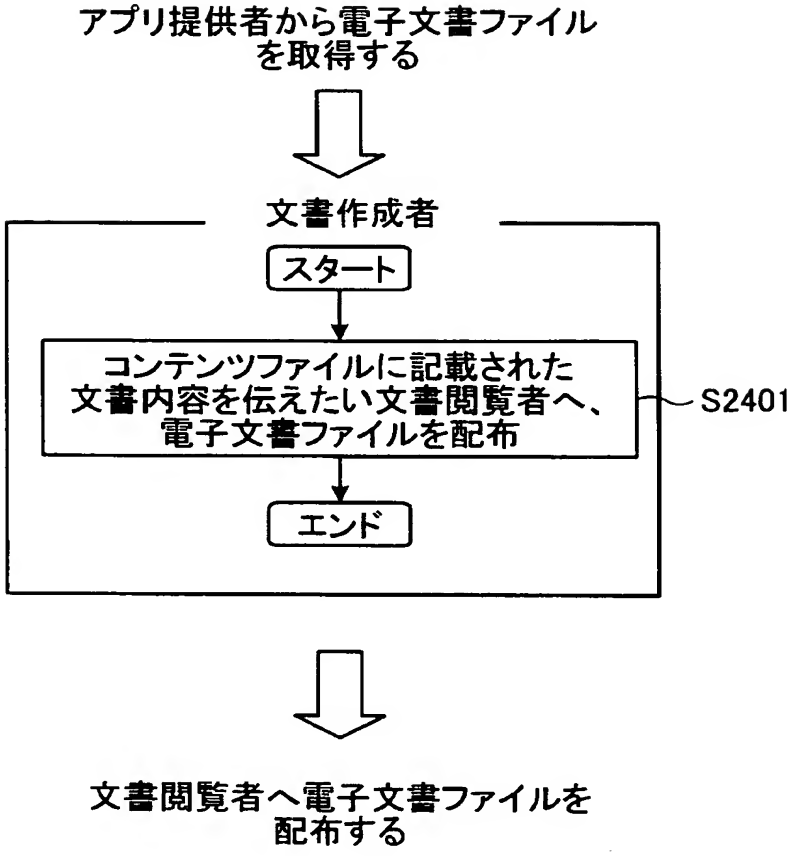
【図２２】


アプリ提供者から電子文書ファイル
を取得する
文書作成者
スタート
S2201
電子文書ファイル内にある動作
プログラムを使用してコンテンツ
ファイルに文書閲覧者へ伝えたい
文書内容を記述
S2202
必要に応じてコンテンツファイル内
にあるコンテンツに対する使用制限
などをコンテンツファイル利用情報
に記述
S2203
閲覧者に
信頼されているか
？
Yes
S2204
電子文書ファイルを
文書閲覧者へ配布
No
S2205
電子文書ファイルをアプリ提供者に
返信
エンド
記述した電子文書ファイルをアプリ提供者
又は文書閲覧者へ配布

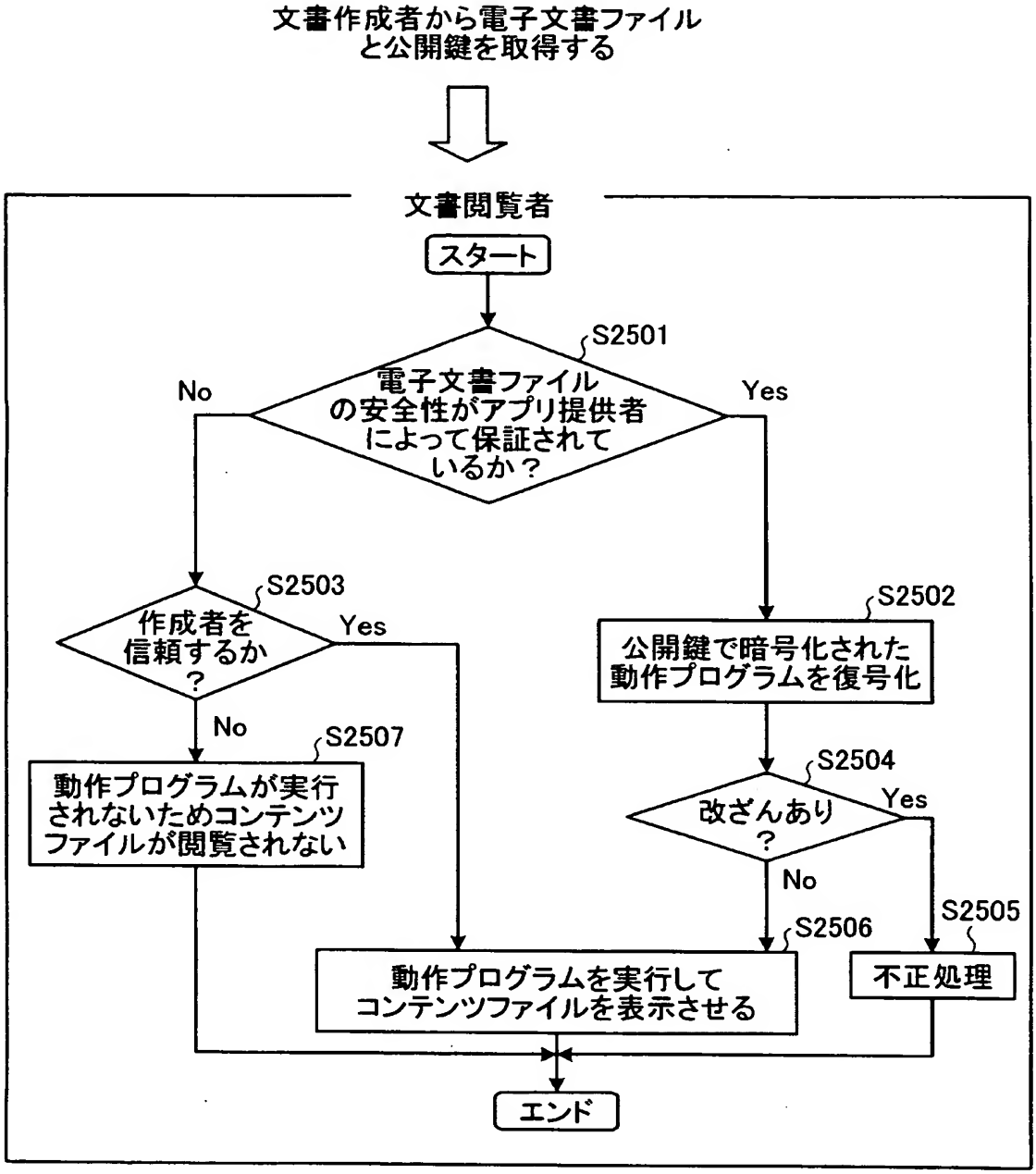
【図２３】


文書作成者から電子文書ファイル
を取得する
アプリ提供者
スタート
電子文書ファイル内の動作プログラム
及び動作プログラム利用情報を削除
S2301
秘密鍵・公開鍵を生成する
S2302
動作プログラム及び動作プログラム
利用情報を新たに生成し、秘密鍵で
暗号化
S2303
動作プログラム利用情報を生成
して、公開鍵情報を保存
S2304
電子文書ファイルに暗号化された動作
プログラム・動作プログラム利用情報
を挿入
S2305
エンド
作成した電子文書ファイルを文書
作成者へ返信する

【図２４】


アプリ提供者から電子文書ファイル
を取得する
文書作成者
スタート
コンテンツファイルに記載された
文書内容を伝えたい文書閲覧者へ、
電子文書ファイルを配布
S2401
エンド
文書閲覧者へ電子文書ファイルを
配布する

【図２５】


文書作成者から電子文書ファイル
と公開鍵を取得する
文書閲覧者
スタート
S2501
電子文書ファイル
の安全性がアプリ提供者
によって保証されて
いるか？
No
Yes
S2503
作成者を
信頼するか
？
Yes
No
S2507
動作プログラムが実行
されないためコンテンツ
ファイルが閲覧されない
S2502
公開鍵で暗号化された
動作プログラムを復号化
S2504
改ざんあり
？
Yes
No
S2505
不正処理
S2506
動作プログラムを実行して
コンテンツファイルを表示させる
エンド

【書類名】要約書
【要約】
【課題】文書作成アプリケーションに電子文書の安全性を保証させることによって、電子文書の作成者を確認することなくても、有害な電子文書の閲覧を防止し、電子文書閲覧の安全性を図ること。
【解決手段】本発明の電子文書ファイルは、文書作成者が作成する文書内容（コンテンツ）を保存するコンテンツファイル１０１，１０２と、コンテンツファイル１０１，１０２内に保存されたコンテンツ情報に対する利用情報を含むコンテンツファイル利用情報１０３と、アプリ提供者（動作プログラム提供者）から提供されるプログラムを含む動作プログラム１０４，１０５，１０６と、動作プログラム１０４，１０５，１０６に対する使用制限や使用方法、動作プログラム１０４，１０５，１０６の改ざん検証をする際に利用する動作プログラム１０４，１０５，１０６の特徴量や復号化情報などを含む動作プログラム利用情報１０７と、を含み構成される。
【選択図】　　図１

特願２００３－２９９１３５

出　願　人　履　歴　情　報

識別番号　　［０００００６７４７］

１．変更年月日　　２００２年　５月１７日
［変更理由］　　住所変更
住　所　　東京都大田区中馬込１丁目３番６号
氏　名　　株式会社リコー